

Classification of Functional Data with k -Nearest-Neighbor Ensembles by Fitting Constrained Multinomial Logit Models

Karen Fuchs^{a,b,1,*}, Wolfgang Pößnecker^{b,1}, Gerhard Tutz^b

^a*Siemens AG, CT RDA SII CPS-DE, Otto-Hahn-Ring 6, D-81739 Munich, Germany*

^b*Department of Statistics, Seminar of Applied Stochastics, Ludwig-Maximilians-Universität München, Akademiestr. 1, D-80799 Munich, Germany*

Abstract

During the last decades, many methods for the analysis of functional data including classification methods have been developed. Nonetheless, there are issues that have not been addressed satisfactorily by currently available methods, as, for example, feature selection combined with variable selection when using multiple functional covariates. In this paper, a functional ensemble is combined with a penalized and constrained multinomial logit model. It is shown that this synthesis yields a powerful classification tool for functional data (possibly mixed with non-functional predictors), which also provides automatic variable selection. The choice of an appropriate, sparsity-inducing penalty allows to estimate most model coefficients to exactly zero, and permits class-specific coefficients in multi-class problems, such that feature selection is obtained. An additional constraint within the multinomial logit model ensures that the model coefficients can be considered as weights. Thus, the estimation results become interpretable with respect to the discriminative importance of the selected features, which is rated by a feature importance measure. In two application examples, data of a cell chip used for water quality monitoring experiments and phoneme data used for speech recognition, the interpretability as well as the selection results are examined. The classification performance is compared to various other classification approaches which are in common use.

Keywords: Functional Data Classification, Variable Selection, Feature Selection, Penalized Multinomial Logit Model, Nearest Neighbor Ensembles, Cell Chip Data, Phoneme Data

☆ Please see the arXiv comments for information about supplementary material.

*Corresponding author

Email addresses: karenfuchs@gmx.de (Karen Fuchs),
Wolfgang.Poessnecker@stat.uni-muenchen.de (Wolfgang Pößnecker),
gerhard.tutz@stat.uni-muenchen.de (Gerhard Tutz)

¹contributed equally

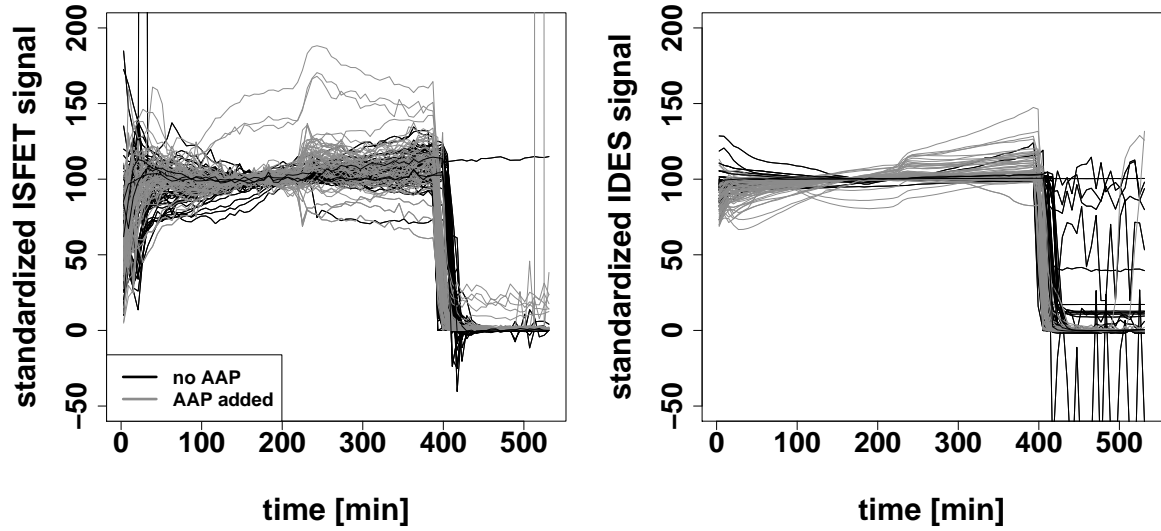


Figure 1: A total of $n = 120$ standardized ISFET- and IDES-signals, recorded over time. Gray scales refer to the two classes of the discrimination task, with gray curves representing measurements with, and black curves measurements without the test substance AAP.

1. Introduction

Functional data analysis is an active field of research, and various methods for classification have been introduced by now. The term “functional” generally refers to the predictors used for classification, which are curves $x_i(t)$, with t from a domain \mathbb{D} , typically an interval from \mathbb{R} . Functional covariates are infinite dimensional in theory, but, due to the limitations in data generation and processing, high dimensional in practice. In Ramsay and Silverman (2005), a thorough introduction to functional data analysis and its most established modeling approaches can be found. Recent research on classification with functional predictors include James (2001), Müller and Stadtmüller (2005), Rossi and Villa (2006), Epifanio (2008), Zhu et al. (2010), Delaigle and Hall (2012), Gertheiss et al. (2013), or Nguyen et al. (2016).

Examples for functional predictors are given by the data motivating our approach. These are the phoneme data introduced by Hastie et al. (1995) and cell chip data. The cell chip measurements are shown in Figure 1. There are, among others, ISFET and IDES sensors implemented on a chips’ surface, whose signals are recorded concurrently over time. Each sensor type captures aspects of the metabolism of a cell monolayer that is grown on the chips’ surface, see also Section 3 for further details. Since the signals represent physiological parameters of the cells’ metabolism, and the latter is a continuous process over time, the sensor signals can be taken as realizations of a continuous function.

The classification task in this data set results from disturbing the cells' habitual environment: While they are usually kept in nutrient medium, one can add test substances, as, for example, paracetamol (short: AAP), to this medium and monitor the cells' reactions. A functional two-class discrimination task is then given by comparing ISFET- and IDES-curves of measurements with (gray curves in Figure 1) and without (black curves) adding AAP to the nutrient medium. Such discrimination tasks occur in the context of environmental monitoring, for example, water quality monitoring, where signals from clean samples are to be distinguished from polluted ones.

We introduce a novel interpretable feature selection method for classifying functional data, where the estimation of a functional k -nearest-neighbor ensemble (k NNE) is carried out by a penalized and constrained multinomial logit model (cMLM). The functional k NNE is set up by a large number of posterior probabilities for class membership that represent the ensemble members. Each posterior probability depends on the k neighbors relative to the observation that is to be classified, as well as on a chosen semi-metric. As was shown by Ferraty and Vieu (2006), semi-metrics are a suitable mathematical formulation to capture certain characteristics of a curve or function. Thus, by using adequate semi-metrics, a large variety of curve features, for example the curves' maxima or their mean values, can be included in the k NNE. The ensemble combines its members in a linear way, and each posterior probability, i.e. ensemble member, is multiplied by an unknown coefficient which has to be determined. This yields the possibility to let members have differing importances. A (multinomial) logit link is then applied on these ensembles to obtain a classification model.

The ensemble coefficients are estimated by a cMLM, combined with a Lasso-type penalty. Lasso penalization allows for sparse results, estimating most coefficients equal to zero, as shown in Tibshirani (1996). Various Lasso-type penalties, suited for MLM, have been developed (see Argyriou et al., 2007, Simon et al., 2013, Chen and Li, 2013, Vincent and Hansen, 2014, among others). Apart from the standard global Lasso penalty (Tibshirani, 1996), we also employ the category-specific Lasso and the categorically structured (CATS) Lasso penalty (Tutz et al., 2015) for multi-class applications. The main advantage of using a Lasso-type penalty is the sparse estimation result enabling feature selection. Additionally we extend an ordinary penalized MLM such that a non-negativity constraint is put on the ensemble coefficients. This constraint ensures a proportional interpretability between the selected features with regard to the data background. As an example, consider two potentially interesting features in the above cell chip data, the step most curves exhibit around 210 minutes, and the distances between measurement curves in the region from about 210 to 400 minutes. By including corresponding semi-metrics in the k NNE and estimating the assigned coefficients by the penalized cMLM, one can decide whether the

step or rather the curve distances yield more discriminative power.

As mentioned, several functional classification approaches have been developed in recent years. Some of them also use ensemble methods that combine scores or features of some kind in a linear combination. Among others, this kind of ensemble was used in Athitsos and Sclaroff (2005), Preda et al. (2007), or Araki et al. (2009). Semi-metrics in the context of functional data classification were used for example in Ferraty and Vieu (2003) and Alonso et al. (2012). All these approaches use a pre-defined and limited number of features that are given as input to a classification algorithm, and usually use the observed curve across its whole domain. This also holds for Matsui (2014), who use a functional logistic regression model to differ between functional observations of different classes, focusing on variable selection. In contrast, our approach can handle a very large number of semi-metrics, i.e. features, in the functional k NNE. This includes taking only specific parts of the curves into account, which might be more significant in certain data situations than using the whole curve, as the above cell chip data suggests. Additionally, for most functional classification methods, the extension to multiple functional covariates is not straight forward, and the estimation results are not interpretable with regard to the data background. Interpretability also is the most weak point in the method by Möller et al. (2016), who built random forests out of summary quantities calculated from (randomly chosen) curve segments. The approach presented here offers both, the potential inclusion of multiple functional (and non-functional) covariates as well as interpretability. These two advantages are also present in the method of Fuchs et al. (2015), who apply a similar ensemble on functional covariates, but estimate the ensemble coefficients by means of Brier score minimization. This implies a more restrictive constraint on the coefficients than is necessary with the estimation via a penalized cMLM. Further advantages that arise from the use of a MLM are that MLM can readily be adapted to ordinal data, and estimation is fast and stable. The penalty might be chosen with respect to probable interrelations between the coefficients, such as the CATS Lasso penalty.

The remainder of this manuscript is organized as follows: In Section 2, our approach is introduced in detail. In Sections 3 and 4, the performance of the presented approach is analysed with respect to the cell chip and phoneme data sets, and is compared to other (functional) classification approaches. The manuscript closes with a discussion of the results and further possible extensions concerning our method in Section 5. We provide exemplarily code as well as the respective data to make the application results fully reproducible.

2. Method

In this section, the proposed approach is given in detail. First, the functional kNN ensemble setup is described, with a focus on the basic design and the semi-metrics that capture specific curve features. The following Section 2.2 introduces penalized cMLM and various Lasso-type penalties. Section 2.3 outlines details of the estimation of our penalized cMLM model. In Section 2.4, a short description of alternative classification techniques is given, and the Brier score and misclassification rate are defined as performance measures to explore the predictive capability of the single methods. An additional feature importance measure is introduced, allowing for the proportional interpretability of our method's selection results.

2.1. Functional nearest neighbor ensembles

Let $x_i(t)$, $i = 1, \dots, n$, be a set of curves, i.e. functional predictors, with t from a domain $\mathbb{D} \subseteq \mathbb{R}$. Assume that each curve belongs to one of G different classes, denoted by $y_i \in \{1, \dots, G\}$. Further, let $x_i^{(a)}(t)$ denote the a th derivative of the functional covariate $x_i(t)$, and $d(x_i^{(a)}(t), x_j^{(a)}(t))$ a semi-metric of (the derivatives of) two functional covariates $x_i^{(a)}(t)$, $x_j^{(a)}(t)$, $i \neq j$.

Translating curve features into probabilities via semi-metrics

A simple functional k -nearest-neighbor ensemble can be specified through a set of various semi-metrics $d_l(\cdot, \cdot)$, where $l = 1, \dots, p$ is the index for the different semi-metrics. Each semi-metric is chosen such that it extracts a particular feature from the curves. Apart from the semi-metrics that were already mentioned in the Introduction, representing the step or distances in the cell chip data, further examples are the covariates' maxima or curvatures. Since some curve characteristics might be amplified after derivation, we use the derivatives of the functional predictors as well as the original curves. Thus, the nearest neighbor ensemble to be considered uses semi-metrics $d_l(x_i^{(a)}(t), x_j^{(a)}(t))$, applied on the quantities $x_i^{(a)}(t)$, $x_j^{(a)}(t)$, $i \neq j$. The whole set of semi-metrics we use is given in Table 1.

After having defined adequate semi-metrics, possibly with respect to expert knowledge about the data at hand, the corresponding semi-metric dependent neighborhoods can be defined. With respect to a generic or a new observation $(y^*, x^*(t))$ and a specific semi-metric $d_l(\cdot, \cdot)$, the learning sample $(y_i, x_i^{(a)}(t))$, $i = 1, \dots, n$ is ordered such that

$$d_l(x^{*(a)}(t), x_{(1)}^{(a)}(t)) \leq \dots \leq d_l(x^{*(a)}(t), x_{(k)}^{(a)}(t)) \leq \dots \leq d_l(x^{*(a)}(t), x_{(n)}^{(a)}(t)).$$

Hence, the $x_{(i)}^{(a)}(t)$ are the curves (or their derivatives) from the learning sample, ordered by their distance to $x^{*(a)}(t)$ as measured by the semi-metric $d_l(\cdot, \cdot)$, so that one can define a neighborhood of the k nearest neighbors of $x^{*(a)}(t)$ by

Semi-metric	Formula	The semi-metric focuses on...
$d^{Eucl}(x_i^{(a)}(t), x_j^{(a)}(t))$	$\sqrt{\int_{\mathbb{D}} \left(x_i^{(a)}(t) - x_j^{(a)}(t) \right)^2 dt}$... the absolute distance of two curves (or their derivatives).
$d_{\tau}^{Scan}(x_i^{(a)}(t), x_j^{(a)}(t))$	$\sqrt{\int_{\mathbb{D}} \left(\phi_{\tau}(t) \left(x_i^{(a)}(t) - x_j^{(a)}(t) \right) \right)^2 dt}$ with $\tau \in \mathbb{D}$... the absolute distances of weighted profiles of the original curves (or their derivatives), centered around τ .
$d_{\mathbb{D}_{small}}^{ShortEucl}(x_i^{(a)}(t), x_j^{(a)}(t))$	$\sqrt{\int_{\mathbb{D}_{small}} \left(x_i^{(a)}(t) - x_j^{(a)}(t) \right)^2 dt}$... the absolute distance on a limited part of the domain of definition $\mathbb{D}_{small} \subset \mathbb{D}$ of two curves (or their derivatives).
$d^{Mean}(x_i^{(a)}(t), x_j^{(a)}(t))$	$\left \int_{\mathbb{D}} x_i^{(a)}(t) dt - \int_{\mathbb{D}} x_j^{(a)}(t) dt \right $... the similarity of mean values of the whole curves (or their derivatives).
$d^{relAreas}(x_i^{(a)}(t), x_j^{(a)}(t))$	$\left \frac{\int_{\mathbb{D}_1} x_i^{(a)}(t) dt}{\int_{\mathbb{D}_2} x_i^{(a)}(t) dt} - \left \frac{\int_{\mathbb{D}_1} x_j^{(a)}(t) dt}{\int_{\mathbb{D}_2} x_j^{(a)}(t) dt} \right \right $... the similarity of the relation of areas on parts of the domain of definition, $\mathbb{D}_1, \mathbb{D}_2 \subset \mathbb{D}$.
$d_{bo}^{Jump}(x_i(t), x_j(t))$	$\left x_i(t_b) - x_i(t_o) \right - \left x_j(t_b) - x_j(t_o) \right $... the similarity of jump heights at points $t_b, t_o \in \mathbb{D}$.
$d^{Max}(x_i^{(a)}(t), x_j^{(a)}(t))$	$\left \max \left(x_i^{(a)}(t) \right) - \max \left(x_j^{(a)}(t) \right) \right $... the difference of the curves' (or their derivatives') global maxima.
$d^{Min}(x_i^{(a)}(t), x_j^{(a)}(t))$	$\left \min \left(x_i^{(a)}(t) \right) - \min \left(x_j^{(a)}(t) \right) \right $... the difference of the curves' (or their derivatives') global minima.
$d^{Points}(x_i^{(a)}(t), x_j^{(a)}(t))$	$\frac{1}{E} \sum_{e=1}^E \left x_i^{(a)}(t_e) - x_j^{(a)}(t_e) \right $... the differences at certain observation points t_e (also called "points of impact").

Table 1: Semi-metrics used to set up the k -nearest-neighbor ensemble.

$$\mathcal{N}_l^k(x^{*(a)}(t)) = \{x_j^{(a)}(t) : d_l(x^{*(a)}(t), x_j^{(a)}(t)) \leq d_l(x^{*(a)}(t), x_{(k)}^{(a)}(t))\}.$$

Let $1(\cdot)$ denote the indicator function. Then a single ensemble member is given by the estimated posterior probability w_{gl} that $x^{*(a)}(t)$ is from class g ,

$$w_{gl} = \frac{1}{k} \sum_{\{j: x_j^{(a)}(t) \in \mathcal{N}_l^k(x^{*(a)}(t))\}} 1(y_j = g).$$

It is determined by the number of nearest neighbors k , the semi-metric $d_l(\cdot, \cdot)$, and the order of the derivative a . For a specific observation $(y_i, x_i^{(a)}(t))$ from the learning sample, the posterior probabilities are obtained by a leave-one-out procedure, which is formally given by

$$w_{igl} = \frac{1}{k} \sum_{\{j \neq i: x_j^{(a)}(t) \in \mathcal{N}_l^{k+1}(x_i^{(a)}(t))\}} 1(y_j = g), \quad i = 1, \dots, n.$$

Simple functional nearest neighbor ensembles

The observed classes y_i from the previous section are considered as realizations of random variables Y_i that take on values in $\{1, \dots, G\}$. Since the methodology uses p semi-metrics, it comprises p posterior probabilities w_{igl} for each observation i and class g . These single posterior probabilities can be combined in a linear combination to obtain a simple ensemble model for the overall posterior probability that observation $x_i(t)$ is from class g , given by

$$\pi_{ig} = \sum_{l=1}^p w_{igl} c_l, \tag{1}$$

where c_l are weights that have to be estimated and must satisfy

$$\sum_{l=1}^p c_l = 1 \quad \text{and} \quad c_l \geq 0 \quad \forall l. \tag{2}$$

Similar to the k -nearest-neighbor classifier for multivariate data, each observation is assigned to the class of highest posterior probability, i.e. $\hat{y}_i = \max_g (\pi_{ig})$.

In addition to the various semi-metrics used in Ensemble (1), we also use different sizes for the number of nearest neighbors $k \in \mathcal{K}_{nN} = \{k_1, \dots, k_M\}$ as well as varying orders of derivation $a \in \{a_1, \dots, a_O\}$. The definitions of the above neighborhood $\mathcal{N}_l^k(x^{*(a)}(t))$ and single posterior probability estimates w_{igl} are adapted accordingly. This means that

from now on, the index l does not refer to a single semi-metric, but represents an ensemble member determined by a unique tuple $\{d(\cdot, \cdot), a, k\}$ of a specific semi-metric $d(\cdot, \cdot)$, a number of nearest neighbors k and an order of derivative a .

If multiple functional covariates are observed, it might be necessary to include covariate type-specific semi-metrics into the ensemble. For instance this might arise if the covariate types originate from different domains, as for example time and wavelengths. Another situation is given in our cell chip application in Section 3, where two different sensors measure different physiological parameters. With R functional predictor types, one has observations $(y_i, x_{i1}(t), \dots, x_{ir}(t))$, $r = 1, \dots, R$, such that each neighborhood, and with that each ensemble member and ensemble coefficient, additionally depends on the covariate type.

Let q , R , M , and O denote the numbers of semi-metrics, covariate types, nearest neighbors, and orders of derivation used. Then, the ensemble comprises a total number of $p = q \cdot R \cdot M \cdot O$ members.

One way to estimate Ensemble (1) with respect to Constraint (2) is to optimize some loss function like the Brier score. Such an estimation approach does not allow for category-specific ensemble coefficients, and the extension to ordinal classes is not self-evident. In the next Section, we propose an alternative estimation technique.

2.2. The penalized and constrained multinomial logit model

Alternatively to loss functions, the estimation of Ensemble (1) can be performed via a multinomial logit model, yielding sparse and interpretable results if being penalized and constrained adequately. To illustrate this, for $g = 1, \dots, G-1$, let $v_{igl} = (w_{igl} - w_{iG})$ denote the differences in posterior probability between classes. For a more compact notation, let $\mathbf{v}_{ig} = (v_{ig1}, \dots, v_{igp})^T$ and $\mathbf{v}_i = (\mathbf{v}_{i1}^T, \dots, \mathbf{v}_{i,G-1}^T)^T$. We consider, for $g = 1, \dots, G-1$, the following constrained multinomial logit model:

$$\begin{aligned} P(Y_i = g | \mathbf{v}_i) &= \pi_{ig} \\ &= \frac{\exp(\mathbf{v}_{ig}^T \mathbf{c})}{1 + \sum_{s=1}^{G-1} \exp(\mathbf{v}_{is}^T \mathbf{c})} \\ &= \frac{\exp\left(\sum_{l=1}^p v_{igl} c_l\right)}{1 + \sum_{s=1}^{G-1} \exp\left(\sum_{l=1}^p v_{isl} c_l\right)} \quad \text{s.t. } c_l \geq 0 \ \forall l. \end{aligned} \quad (3)$$

The probability for the “reference class” G is trivially given by

$$\pi_{iG} = 1 - \sum_{s=1}^{G-1} \pi_{is} = \frac{1}{1 + \sum_{s=1}^{G-1} \exp(\mathbf{v}_{is}^T \mathbf{c})}.$$

The novelty in Model (3) relative to an ordinary MLM is the constraint on the model coefficients. The restriction to values equal or above zero allows a proportional interpretation of the coefficients’ values with respect to the data background, and thus yields a relevant improvement.

Model (3) can also be rewritten in terms of log odds with regard to the reference class G as follows:

$$\begin{aligned} \log \left(\frac{P(Y_i = g | \mathbf{v}_i)}{P(Y_i = G | \mathbf{v}_i)} \right) &= \mathbf{v}_{ig}^T \mathbf{c} = \sum_{l=1}^p v_{igl} c_l \\ &= \sum_{l=1}^p (w_{igl} - w_{iGl}) c_l \quad \text{s.t. } c_l \geq 0 \quad \forall l \end{aligned}$$

for $g = 1, \dots, G - 1$. If $w_{igl} > w_{iGl}$ with $c_l \neq 0$, then the logit will change in favor of class g over G accordingly, and vice versa for $w_{igl} < w_{iGl}$. Hence, when classes show differences in their posterior probabilities which are based on a particular curve feature l , this information gets automatically translated into a change of the overall class probability. Given the difference building on the posterior probabilities, which in our model have the role of covariates, the nonnegativity constraint that is imposed on the parameters c_l allows to interpret them as weights that reflect the importance of the different curve features for the classification. From a more technical point of view, the w_{igl} are class-specific covariates, and using their difference to a reference class (here G) is necessary to make the MLM identifiable. The use of these differences also arise naturally when the MLM is motivated via latent utility maximization as shown in McFadden (1973).

Since the linear predictors $\mathbf{v}_{ig}^T \mathbf{c}$ are transformed to class probabilities via the multinomial logit link, feasible estimates $\hat{\pi}_{ig} \in [0, 1]$ are guaranteed for any parameter estimate $\hat{\mathbf{c}}$. This holds for both the observed data as well as in prediction, so that Model (3) can

easily be extended to more flexible, class-specific weights c_{gl} :

$$\begin{aligned} P(Y_i = g | \mathbf{v}_i) &= \frac{\exp(\mathbf{v}_{ig}^T \mathbf{c}_g)}{1 + \sum_{s=1}^{G-1} \exp(\mathbf{v}_{is}^T \mathbf{c}_s)} \\ &= \frac{\exp\left(\sum_{l=1}^p v_{igl} c_{gl}\right)}{1 + \sum_{s=1}^{G-1} \exp\left(\sum_{l=1}^p v_{isl} c_{sl}\right)} \quad \text{s.t. } c_{gl} \geq 0 \ \forall g, l. \end{aligned} \quad (4)$$

It was shown in Gertheiss and Tutz (2009) that $\hat{\pi}_{ig} \in [0, 1]$ cannot be guaranteed in prediction if one uses class-specific weights c_{gl} in the simple linear ensemble approach (1). Hence, the modeling option (4) is a major advantage of our cMLM approach since it allows a more flexible model for classification tasks in problems with more than two classes. The analysis of the phoneme data in Section 4 illustrates this advantage on real data.

Penalization for the constrained MLM

Depending on the data at hand and the choices of the sets of predictors, semi-metrics, the number of different neighborhood sizes and the number of different orders of co-variate derivatives used, the number p of coefficients that has to be estimated can easily reach orders of 10^2 to 10^5 . To regularize the estimates and to find an interpretable set of curve features that explains the curves' class memberships, we propose to use penalized estimation with variable selection penalties.

In the following, let $\log(L(\mathbf{c}))$ denote the log-likelihood of Models (3) or (4), let $J(\mathbf{c})$ denote a penalty term and let $\lambda \geq 0$ denote a tuning parameter that controls the strength of the penalization. For the model from (3) that uses global weights on all curve features, a suitable penalty is given by the Lasso of Tibshirani (1996), yielding

$$\hat{\mathbf{c}} = \underset{\mathbf{c}}{\operatorname{argmax}} \left(\log(L(\mathbf{c})) - \lambda J(\mathbf{c}) = \log(L(\mathbf{c})) - \lambda \sum_{l=1}^p |c_l| \quad \text{s.t. } c_l \geq 0 \ \forall l \right). \quad (5)$$

Due to its mathematical properties, the Lasso penalty $J(\mathbf{c}) = \sum_{l=1}^p |c_l|$ induces sparse estimates for suitably chosen λ , that is, one obtains $\hat{c}_l = 0$ for many l , yielding selection of curve features.

For the model from (4) with class-specific weights, the Lasso estimates are given by

$$\hat{\mathbf{c}} = \underset{\mathbf{c}}{\operatorname{argmax}} \left(\log(L(\mathbf{c})) - \lambda \sum_{l=1}^p \sum_{g=1}^{G-1} |c_{gl}| \quad \text{s.t. } c_{gl} \geq 0 \ \forall g, l \right). \quad (6)$$

In this case, the Lasso approach induces solutions that are sparse on the parameter level,

i.e. it is possible to obtain, for example, $\hat{c}_{1l} > 0$ and $\hat{c}_{2l} = 0$. Since the denominator in (4) is influenced by all parameters, the respective probability $P(Y_i = 2|\mathbf{v}_i)$ would still be influenced by curve feature l .

To obtain a proper selection of curve features, one can adopt the Categorically Structured Lasso approach of Tutz et al. (2015), which computes estimates according to

$$\hat{\mathbf{c}} = \underset{\mathbf{c}}{\operatorname{argmax}} \left(\log(L(\mathbf{c})) - \lambda \sqrt{G-1} \sum_{l=1}^p \sqrt{\sum_{g=1}^{G-1} c_{gl}^2} \quad \text{s.t. } c_{gl} \geq 0 \ \forall g, l \right). \quad (7)$$

The CATS approach treats all parameters that belong to the same curve feature l as one parameter group that is removed from the model jointly. However, here the CATS approach can produce solutions with so-called ‘within-group-sparsity’, which is in stark contrast to the behavior of CATS in the context of Tutz et al. (2015). This phenomenon has technical reasons, see also Equation (8) in the following section.

2.3. Computation of estimates

From a technical point of view, our constrained and penalized MLM is simply an ordinary penalized MLM with class-specific covariates and a constraint. Therefore, our model is covered by the framework of Tutz et al. (2015) except for the nonnegativity constraint on the parameters. Hence, the FISTA algorithm for the estimation of a penalized MLM proposed in Tutz et al. (2015) must be adapted to incorporate this constraint. The core problem to solve is the so-called proximal operator with nonnegativity constraint. For the Lasso penalty on global parameters as in (5), and with $\mathbf{u} \in \mathbb{R}^p$ denoting an arbitrary and generic input vector, this problem has the following form:

$$\mathbf{Prox}_{\text{lasso}}(\mathbf{u}|\lambda) = \underset{\mathbf{c} \in \mathbb{R}_{\geq 0}^p}{\operatorname{argmin}} \left(\frac{1}{2} \|\mathbf{c} - \mathbf{u}\|_2^2 + \lambda \sum_{l=1}^p |c_l| \right).$$

As proven in Jenatton et al. (2011), this problem is solved by simply replacing all negative entries of the input vector with zero, which reduces the problem to the well-studied unconstrained proximal operator. With $[\mathbf{u}]_+ = \max(\mathbf{u}, 0)$ (to be understood entry-wise), one obtains

$$\begin{aligned} \underset{\mathbf{c} \in \mathbb{R}_{\geq 0}^p}{\operatorname{argmin}} \left(\frac{1}{2} \|\mathbf{c} - \mathbf{u}\|_2^2 + \lambda \sum_{l=1}^p |c_l| \right) &= \underset{\mathbf{c} \in \mathbb{R}^p}{\operatorname{argmin}} \left(\frac{1}{2} \|\mathbf{c} - [\mathbf{u}]_+\|_2^2 + \lambda \sum_{l=1}^p |c_l| \right) \\ &= \left(\left[[u_l]_+ - \lambda \right]_+ \right)_{l=1, \dots, p} = \left(\max(\max(u_l, 0) - \lambda, 0) \right)_{l=1, \dots, p}. \end{aligned}$$

An equivalent statement holds for the Lasso penalty on class-specific weights from (6). For the CATS penalty from (7), one obtains for $l = 1 \dots p$

$$\underset{\mathbf{c}_l \in \mathbb{R}_{\geq 0}^{G-1}}{\operatorname{argmin}} \left(\frac{1}{2} \|\mathbf{c}_l - \mathbf{u}_l\|_2^2 + \lambda \sqrt{G-1} \sqrt{\sum_{g=1}^{G-1} c_{gl}^2} \right) = \left[1 - \frac{\lambda \sqrt{G-1}}{\|\mathbf{u}_l\|_2} \right]_+ [\mathbf{u}_l]_+. \quad (8)$$

The corresponding estimation algorithm is implemented in the publicly available R-package MRSP (Pöbnecker, 2015). The nonnegativity constraint on the parameters can be activated by using the argument “nonneg = TRUE”.

2.4. Competing methods and prediction performance measures

To be able to evaluate the prediction performance of the kNNE estimated via penalized cMLM, we compare all results to alternative approaches. Since only few functional classification methods introduced so far come with an implementation, we will also use some multivariate discrimination techniques that are known for their good performance. To this end, we either use the discretized covariates if appropriate, or otherwise compute a functional principal component (FPC) analysis and use the respective scores as inputs. To the best of our knowledge, the choice of the number of components in FPC analysis is still an open problem. An approach used by some researchers (e.g. Hall et al., 2001) is to use those scores that explain a specified percentage of the sample variability. We do the same, choosing the number of scores such that at least 95% variability is explained. The FPC computation is carried out with the `fpca.sc`-function of the R-package `refund` (Di et al., 2009, Crainiceanu et al., 2013, Goldsmith et al., 2013).

In the following, all methods included in the comparison are shortly presented. For details on the single methods, the reader is referred to the literature. The calculations for all methods, including the penalized cMLM, are carried out using the software environment R (R Core Team, 2016) and respective add-on packages. Details on the latter as well as on the choices of parameters, where appropriate, are given in the following. If not stated otherwise, the default parameters are used.

Penalized constrained multinomial logit model (cMLM) The method introduced in Sections 2.1 - 2.3. Before modeling, both the learning as well as the test data set are normalized to a standard deviation of one. The penalty parameter λ is chosen from a predefined grid of values. The model choice bases on the minimization of the mean Akaike information criterion (AIC). The abbreviation *cMLM* is used whenever a global Lasso penalty is employed. The abbreviation *cs cMLM* implies that a

category-specific Lasso penalty is used, *csCATS cMLM* denotes the usage of a categorically structured Lasso penalty. For computation, the add-on package MRSP (Pößnecker, 2015) is used.

k-nearest-neighbor ensemble (*kNN Ensemble*, Fuchs et al., 2015) The ensemble is, apart from the standard deviation’s normalization, identical to the one set up for the cMLM, i.e. Model (1) has to satisfy Constraints (2). The ensemble coefficients are estimated via the Brier score minimization used in the original research article. Software from the respective supplement is used.

Nonparametric functional classification (*NPFC*, Ferraty and Vieu, 2003) We use four semi-metrics the approach offers, namely the Euclidian distance of the a th derivative of the functional covariates (abbreviation *NPFC - deriv*), the Euclidian distance of the Fourier expansions of the functional covariates (abbreviation *NPFC - Fourier*), the multivariate partial least squares regression semi-metric (abbreviation *NPFC - mplsr*), and the functional principal component semi-metric (abbreviation *NPFC - pca*). For details on the single semi-metrics, we refer to Ferraty and Vieu (2006). All semi-metrics require the choice of at least one parameter. These are chosen by minimizing the mean misclassification error of a 10-fold CV. The modeling software can be found at <http://www.math.univ-toulouse.fr/ferraty/SOFTWARES/NPFDA/index.html>.

Functional linear model (*FLM - log*, Ramsay and Silverman, 2005) In the case of a two-class problem, we use a parametric functional model, which is implemented in the *gam*-function of the R package *mgcv* (Wood, 2014). The number of basis functions used for each smooth term has to be chosen. We test a grid $\{3, \dots, 10\}$ and choose the number with minimal mean misclassification rate in the test data.

Functional random forests (*fRF*, Möller et al., 2016) The software has kindly been provided by the authors of Möller et al. (2016). The classification is carried out by the function *FuncRandomForest*. Differing from the default call, the variables importance and overlap are set to TRUE to have access to the variable importance measure of the method and achieve a more flexible interval choice. The model parameters λ and c were each chosen from predefined grids via minimizing the mean misclassification rate of a 5-fold CV.

Linear discriminant analysis (*LDA*, Fisher, 1936, Rao, 1973) We apply *LDA* to the FPC scores, as done by Ramsay and Silverman (2002). *LDA* is implemented in the R package *MASS* (Ripley et al., 2014), function *lda*.

Penalized discriminant analysis (*PDA - cov.*, Hastie et al., 1995) *PDA* was especially designed for high-dimensional and highly correlated covariates (Hastie et al., 1995),

such that it can be applied on the discretized data. The approach is implemented in the R package `mda` (Hastie et al., 2015b), function `fda`.

Multinomial model (*mM*, see e.g. Tutz, 2012) A multinomial logistic regression model is used on the FPC scores. This method is implemented in the `maxent`-function of the R package `maxent` (Jurka and Tsuruoka, 2013).

Support vector machines (*SVM*, Vapnik, 1996) We use the implementation of the R package `e1071` (Meyer et al., 2014) by using the function `svm`, with `probability=TRUE` and default settings else. The *SVM* is applied to both, the discretized data, referred to by *SVM - cov.*, and the FPC scores (*SVM - FPCs*).

Random forests (*RF*, Breiman, 2001) The method is implemented in the R package `randomForest` (Breiman et al., 2012). We used the `randomForest`-function. *RF* are applied to both, the discretized data (*RF - cov.*) and the FPC scores (*RF - FPCs*).

Regularized discriminant analysis (*RDA*, Guo et al., 2007) *RDA* penalizes a *LDA* and was designed for high-dimensional data, thus being applied to the discretized covariates. It is available from the R package `rda` (Guo et al., 2012), function `rda`.

Sparse discriminant analysis (*SDA*, Clemmensen et al., 2011) Another modification of *LDA* is *SDA*, which we apply to the discretized covariates. An implementation can be found in the R package `sda` (Ahdesmaki et al., 2015), function `sda`.

To be able to compare the prediction performances of the competing methods described above, we will use two performance measures. The first is the Brier score operating on the coded response $z_{ig} = 1$ if $y_i = g$ and $z_{ig} = 0$ otherwise,

$$Q = \frac{1}{n_{test}} \frac{1}{G} \sum_{i=1}^{n_{test}} \sum_{g=1}^G (z_{ig} - \pi_{ig})^2,$$

introduced by Brier (1950). n_{test} denotes the sample size of the test data. As shown by Gneiting and Raftery (2007), the Brier score is a proper scoring rule. Moreover, it is the only one that fulfills, up to a positive linear transformation, the properties that Selten (1998) demanded of scoring rules. This implies that, in contrast to, for example, the logarithmic score, the Brier score does not react strongly on small differences between small probabilities, especially probabilities of values (around) zero. Although improper in the above sense, the second performance measure we use is the classical misclassification rate (MCR)

$$MCR = \frac{1}{n_{test}} \sum_{i=1}^{n_{test}} 1(y_i \neq \hat{y}_i).$$

Here, $1(\cdot)$ again denotes the indicator function, y_i denotes the true class of observation $x_i(t)$, and \hat{y}_i the class assigned by the method considered.

In addition to these performance measures we introduce a score that indicates how important the coefficients, estimated by the penalized cMLM, are relative to each other, i.e. which of the features corresponding to the coefficients yields most discriminative power. The score is called the relative feature importance (RFI) measure and yields the percental importance per estimated coefficient \hat{c}_l . It is defined by

$$RFI_l = 100 \cdot \frac{\sum_{g=1}^G \hat{c}_{gl}}{\sum_{g=1}^G \sum_{s=1}^p \hat{c}_{gs}}.$$

In the case of global coefficients, the RFI measure simplifies to $RFI_l = 100 \cdot \frac{\hat{c}_l}{\sum_{s=1}^p \hat{c}_s}$.

3. Application to real world data - cell based sensor chips

Cell based sensor technologies attract much interest in biomechanical engineering, especially in application fields concerned with environmental quality monitoring, see Bohrn et al. (2012) or Kubisch et al. (2012) for some examples and references.

In this study we use cell based chips, which means that the chips' surfaces are covered with a monolayer of a living cell population. There are different kinds of sensors distributed across the chip surface, which record different cell reactions. Five ion-sensitive field-effect transistors (ISFET) measure the pH value of the extracellular medium. A high acidification of the medium correlates with a high metabolic rate of the cells. One interdigitated electrode structure (IDES) is used to draw conclusions about the cell morphology and cell adhesion of the cells on the chip surface (Thedinga et al., 2007, Ceriotti et al., 2007). The signals of each sensor are recorded concurrently over time, representing the above aspects of the cells' metabolism. Since the latter is undoubtedly a continuous process, the sensor signals might be taken for functional data, observed at equidistant, discrete time points. In our study, we use the arithmetic mean of signals of the same type.

For a detection layer, we use chinese hamster lung fibroblast cells due to their stable and reliable growth (Bohrn et al., 2013). Their usual environment is nutrient medium. If the composition of this medium alters, for example by adding some test substance to it, the cells will react accordingly. To evaluate the performance of our approach in a binary classification problem, the cell chip data is restricted to measurements with nutrient medium only, and measurements where 2.5mM paracetamol is added. The data set includes $n = n_0 + n_1 = 120$ measurements per signal type of $Q = 89$ equidistant observation points, $n_0 = 63$ without and $n_1 = 57$ with AAP, depicted in Figure 2.

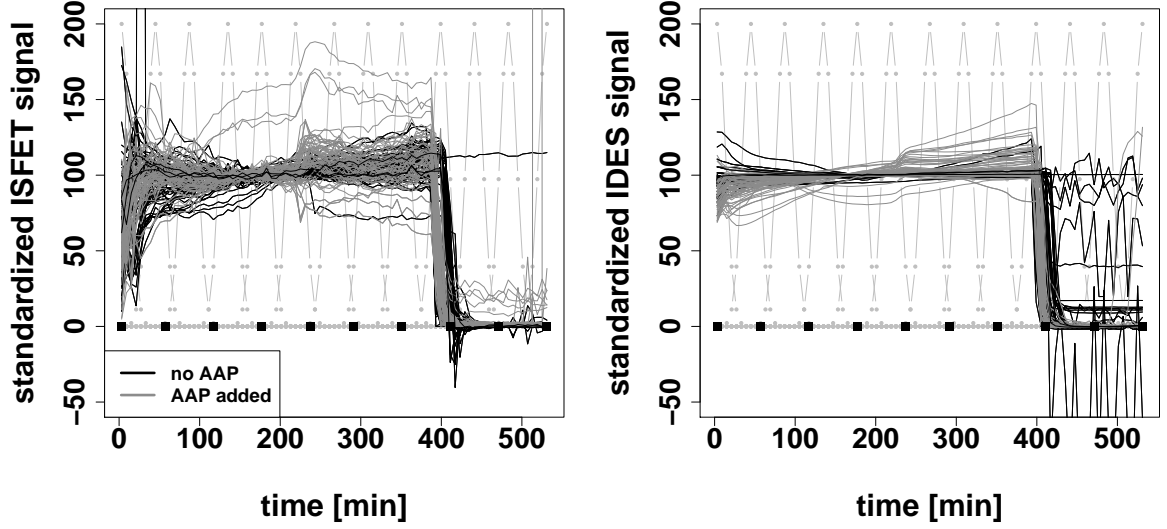


Figure 2: The same $n = 120$ standardized ISFET- and IDES-signals as before. The light gray, dotted lines depict the function $\phi_\tau(t)$ in d_τ^{Scan} used at certain observation points τ , the impact points t_e used in d^{Points} are depicted as black boxes.

The measurements can be divided into three phases: the first corresponds to an acclimatisation phase with medium flowing over the cells to let them adapt to the system and get stable physiological signals. At the second phase from about 220 minutes on, the test substance (AAP) reaches the cells. From about 400 minutes on, 0.2% Triton X-100 is added to the medium in the last measurement phase, removing the cells from the chip surface. This last step is necessary to obtain a negative control. After the cells have adapted to their environment, shortly before the test substance is applied, one expects the cells to exhibit 100% viability. Thus, all signals were standardized in such a way that, at about 170 minutes, the signals have a value of 100.

Since the ISFET- and the IDES-signals originate from different biochemical processes and measurement principles, it is adequate to treat them as two functional covariate types, such that the modeling implies multiple covariates. The character of the single curves, however, is similar, exhibiting all three measurement phases, which is the reason why identical semi-metrics with the same k -nearest-neighbor parameter tuples can be used for both signals.

We use all the semi-metrics listed in Table 1. Further parameters are the numbers of nearest neighbors $k \in \mathcal{K}_{nN} = \{1, 5, 11, 21\}$, and orders of derivation $a \in \{0, 1, 2\}$. The choices of \mathbb{D}_{small} , \mathbb{D}_1 , \mathbb{D}_2 and t_e mainly reflect curve regions where the AAP reaches the cells in phase two and the changeover of phase two and three. For the semi-metric

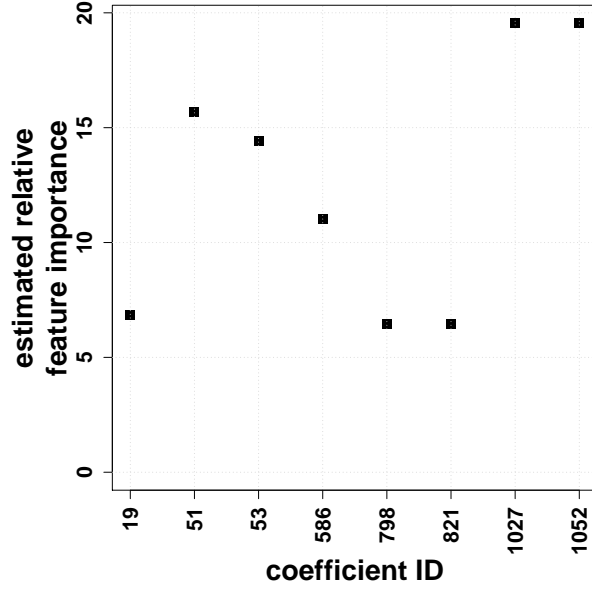


Figure 3: Relative feature importance of the coefficients which have been estimated unequal to zero, as estimated from the whole cell chip data.

$d_{\mathbb{D}_{small}}^{shortEucl}$, one of the intervals $[t_1, t_{35}]$, $[t_{36}, t_{40}]$, $[t_{41}, t_{64}]$, $[t_{65}, t_{69}]$, or $[t_{70}, t_{89}]$ is used for \mathbb{D}_{small} ; for semi-metric d_{bo}^{Jump} , one of the sets $\{36, 39\}$ or $\{65, 68\}$ is used for $\{t_b, t_o\}$; for semi-metric $d^{relAreas}$, \mathbb{D}_1 is one of the intervals $[t_1, t_{35}]$ or $[t_{41}, t_{64}]$, and $\mathbb{D}_2 = [t_{41}, t_{64}]$; for semi-metric d^{Points} , an equidistant grid $t_e = t_{mQ/10}$, $m = 1, \dots, 10$, is used; and the function $\phi_\tau(t) = \frac{300}{\max(\phi_{1,\tau}(t))} \phi_{1,\tau}(t)$ with $\phi_{1,\tau}(t) = \frac{1}{\sqrt{2\pi}\sigma} \exp^{-\frac{1}{2}(\frac{t-\tau}{\sigma})^2}$, $\sigma = 10$ and $\tau \in \{3.120, 45.120, 87.120, 135.120, 177.120, 219.129, 267.129, 309.129, 351.129, 399.140, 441.140, 483.140, 531.140\}$ is used in semi-metric d_τ^{Scan} . In Figure 2, the weight functions $\phi_\tau(t)$ per τ , used in d_τ^{Scan} , are depicted as light gray, dotted lines; the impact points t_e used in d^{Points} are marked by black boxes.

3.1. Results

With the above parameter choices, the ensemble comprises $p = 1248$ ensemble members, i.e. 624 coefficients per signal type that have to be estimated. The respective single posterior probability estimates w_{igl} per curve $x_i(t)$ are calculated as described in Section 2.1. Afterwards, the standard deviation $sd(v_{igl})$ is calculated. If $sd(v_{igl}) \equiv 0 \ \forall i$ for a certain tuple l , the respective tuple is removed from the data set, since it does not contain any information concerning the class. In that way, 80 tuples are removed.

Finally, the penalized cMLM is applied to the whole cell chip data as described in Section 2.2, using a global Lasso penalty and yielding a final number of 1168 coefficients that have to be estimated. The model yielding minimal mean AIC employs a penalty parameter of $\lambda_{Lasso} = 1.9$. The RFI of the coefficient estimates resulting from the whole cell chip data

coefficient ID (estimated RFI values)	parameter tuple
Lasso	
1027 (19.55%)	$\{d^{shortEucl}, a = 2, k = 5\}, \mathbb{D}_{small} = [t_{36}, t_{40}],$ covariate IDES
1052 (19.55%)	$\{d^{shortEucl}, \text{with } x_i(t) \text{ centered}, a = 2, k = 5\},$ $\mathbb{D}_{small} = [t_{36}, t_{40}],$ covariate IDES
51 (15.68%)	$\{d^{Eucl}, a = 0, k = 5\},$ covariate ISFET
53 (14.41%)	$\{d^{shortEucl}, a = 0, k = 5\}, \mathbb{D}_{small} = [t_{36}, t_{40}],$ covariate ISFET
586 (11.03%)	$\{d^{shortEucl}, a = 0, k = 1\}, \mathbb{D}_{small} = [t_1, t_{35}],$ covariate IDES

Table 2: First column: the IDs of the five estimated coefficients that show the largest relative feature importance (RFI) values (in brackets, in decreasing order). Second column: The chosen ensemble coefficients are decoded, with value a indicating the order of derivation and k indicating the number of nearest neighbors used.

set are depicted in Figure 3. For clarity, only the 8 coefficients that have been estimated with values unequal to zero are shown. The tuples corresponding to the five coefficients estimated with the highest RFI values are decoded in Table 2.

To evaluate the prediction performance of our method and to be able to compare it to the other classification methods, we divide the data set randomly 100 times into learning sets comprising 90 curves and test sets of size 30. All competing methods are listed in Table 3, including their abbreviations. They are applied on identical sample sets.

As with the whole data set, the estimation results of our approach per draw are sparse. Overall, only 84 coefficient IDs across all replications were estimated to have values above zero, and most were selected very seldom, see also the exemplifying boxplots in Figure A.9 in Appendix A. The coefficient ID's with the five highest estimated mean RFI values are 1027, 1052, 586, 597, and 1043, partly overlapping with the coefficients that have been estimated from the whole data listed in Table 2. The tuples corresponding to the unlisted IDs are $597 \triangleq \{d^{Points}, a = 0, k = 1\}$ with covariate IDES and $1043 \triangleq \{d^{Scan}, a = 2, k = 5\}$ with $\tau = 219.13$ and covariate IDES. From the selected coefficients, one can conclude that both signal types, ISFET- as well as IDES-signals, imply discriminative information. Thus, the classification task is fulfilled without variable selection. Concerning feature selection, most of the selected coefficient IDs, i.e. parameter tuples, include the curve region around 220 minutes. This is reasonable since, at this time, the AAP reaches the cells. Coefficient 586, representing a tuple including the first measurement phase of the IDES-signals, seems also a sensible choice, since many of the IDES curves show a

Method	Abbreviation	R function used (package-name)
Constrained multinomial logit model (global Lasso penalty)	cMLM	see Pöbnecker (2015) and the online supplement
Constrained multinomial logit model (category- specific Lasso penalty)	cs cMLM	see above
Constrained multinomial logit model (category- specific CATS penalty)	csCATS cMLM	see above
Functional k -nearest- neighbor ensemble	kNN Ensemble	see online supplement in Fuchs et al. (2015)
Nonparametric functional classification (NPFC)	NPFC-deriv	funopadi.knn.lcv (http://www.math.univ-toulouse.fr/staph/npfda/)
NPFC	NPFC-Fourier	see above
NPFC	NPFC-mplsr	see above
NPFC	NPFC-pca	see above
Functional linear model	FLM-log	gam (mgcv)
Functional random forests	fRF	FuncRandomForest (pro- vided by the authors of Möller et al. (2016))
Linear discriminant analysis	LDA	lda (MASS)
Penalized discriminant analysis	PDA-cov.	fda (mda)
Multinomial model	mM	maxent (maxent)
Support vector classifiers	SVM-FPCs	svm (e1071)
Support vector classifiers	SVM-cov.	see above
Random forests	RF-cov.	randomForest (randomForest)
Random forests	RF-FPCs	see above
Regularized discriminant analysis	RDA	rda (rda)
Shrinkage discriminant analysis	SDA	sda (sda)

Table 3: List of all methods included in the comparison.

class-dependent slope. In conclusion, the feature selection of our penalized cMLM agrees very well with background knowledge of the cell chip data.

Figure 4 gives the test results for all classification approaches on basis of 100 modeling replications. The upper panel shows boxplots of the Brier score across all draws (for *RDA*, the estimated probabilities were not accessible), the lower panel shows boxplots of the misclassification rates. Obviously, the penalized cMLM outperforms all other methods in terms of both performance measures. Thus, the penalized cMLM is a very attractive choice for this discrimination task.

4. Application to real world data - phoneme data

Another data example that became quite popular in functional data classification is the phoneme data introduced in Hastie et al. (1995), available through the R-package ElemStatLearn (Hastie et al., 2015a). The data consists of 4509 log-periodograms, taken as covariates $x_i(t)$, that each are ascribed to one of the five phonemes “aa”, “ao”, “dcl”, “iy”, and “sh”, recorded at 256 frequencies, i.e. observation points. Figure 5 shows five exemplarily curves per phoneme. The goal of this discrimination task is to differ between the log-periodograms, a task arising in the field of speech recognition. The data has, among others, been examined by Hastie et al. (1995), Ferraty and Vieu (2003), Epifanio (2008), and Li and Yu (2008).

The parameter settings for the k -nearest-neighbor ensemble members were chosen arbitrarily due to the absence of relevant background knowledge. We thus use $k \in \mathcal{K}_{nN} = \{1, 5, 11, 21\}$ nearest neighbors and orders of derivation $a \in \{0, 1, 2\}$. One of the intervals $[t_1, t_{17}]$, $[t_{18}, t_{36}]$, $[t_{37}, t_{56}]$, $[t_{57}, t_{76}]$, $[t_{77}, t_{100}]$, or $[t_{30}, t_{65}]$ is used for \mathbb{D}_{small} in semi-metric $d_{\mathbb{D}_{small}}^{shortEucl}$; for semi-metric d_{bo}^{Jump} , one of the sets $\{15, 19\}$, $\{34, 40\}$, $\{54, 58\}$, or $\{74, 78\}$ is used for $\{t_b, t_o\}$; for semi-metric $d^{relAreas}$, \mathbb{D}_1 is one of the intervals $[t_1, t_{17}]$, $[t_{57}, t_{76}]$, or $[t_{30}, t_{65}]$, and $\mathbb{D}_2 = [t_{37}, t_{56}]$; for semi-metric d^{Points} , a grid $t_e \in \{1, 14.42, 27.84, 41.26, 54.68, 68.11, 81.53, 94.95, 108.37, 121.79, 135.21, 148.63, 162.05, 175.47, 188.89, 202.32, 215.74, 229.16, 242.58, 256\}$ is used; and the function $\phi_\tau(t) = \frac{300}{\max(\phi_{1,\tau}(t))} \phi_{1,\tau}(t)$ with $\phi_{1,\tau}(t) = \frac{1}{\sqrt{2\pi}\sigma} \exp^{-\frac{1}{2}(\frac{t-\tau}{\sigma})^2}$, $\sigma = 6$ and $\tau \in \{1, 22, 25, 43, 50, 64, 75, 86, 107, 110, 128, 149, 171, 175, 192, 213, 234, 256\}$ is used for semi-metric d_τ^{Scan} .

4.1. Results

The upper parameter setting results in 816 ensemble coefficients. Using the whole phoneme data, one finds 16 tuples yielding standard deviations $sd(v_{igl}) \equiv 0 \ \forall i$. These tuples are removed such that the ensemble contains $p = 800$ coefficients that have to be

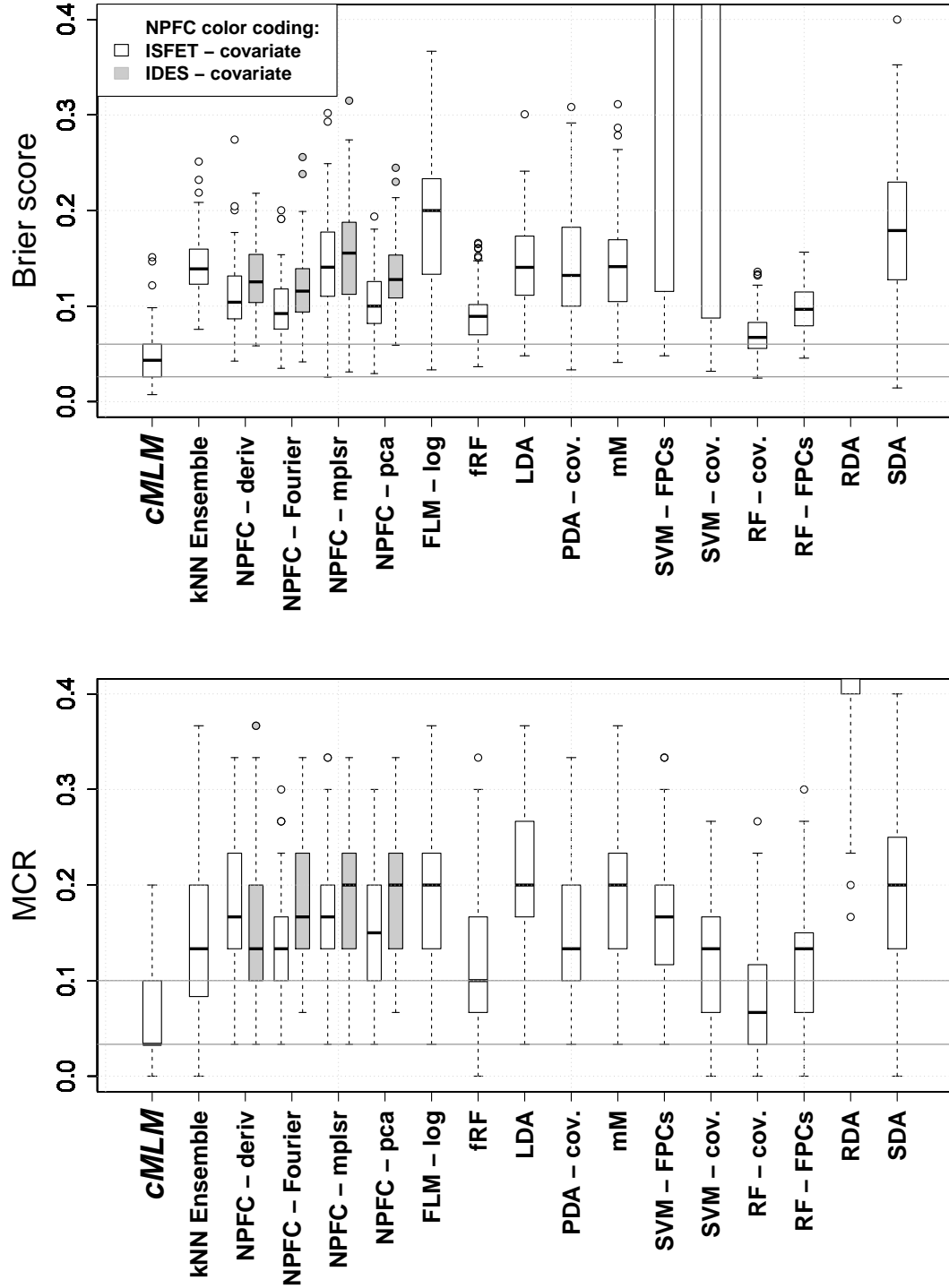


Figure 4: Test results of the cell data for all classification approaches on basis of 100 replications. The upper panel shows the Brier scores, the lower panel the misclassification rates (MCR). The horizontal lines indicate the values of the first and third quartiles of the cMLM-box.

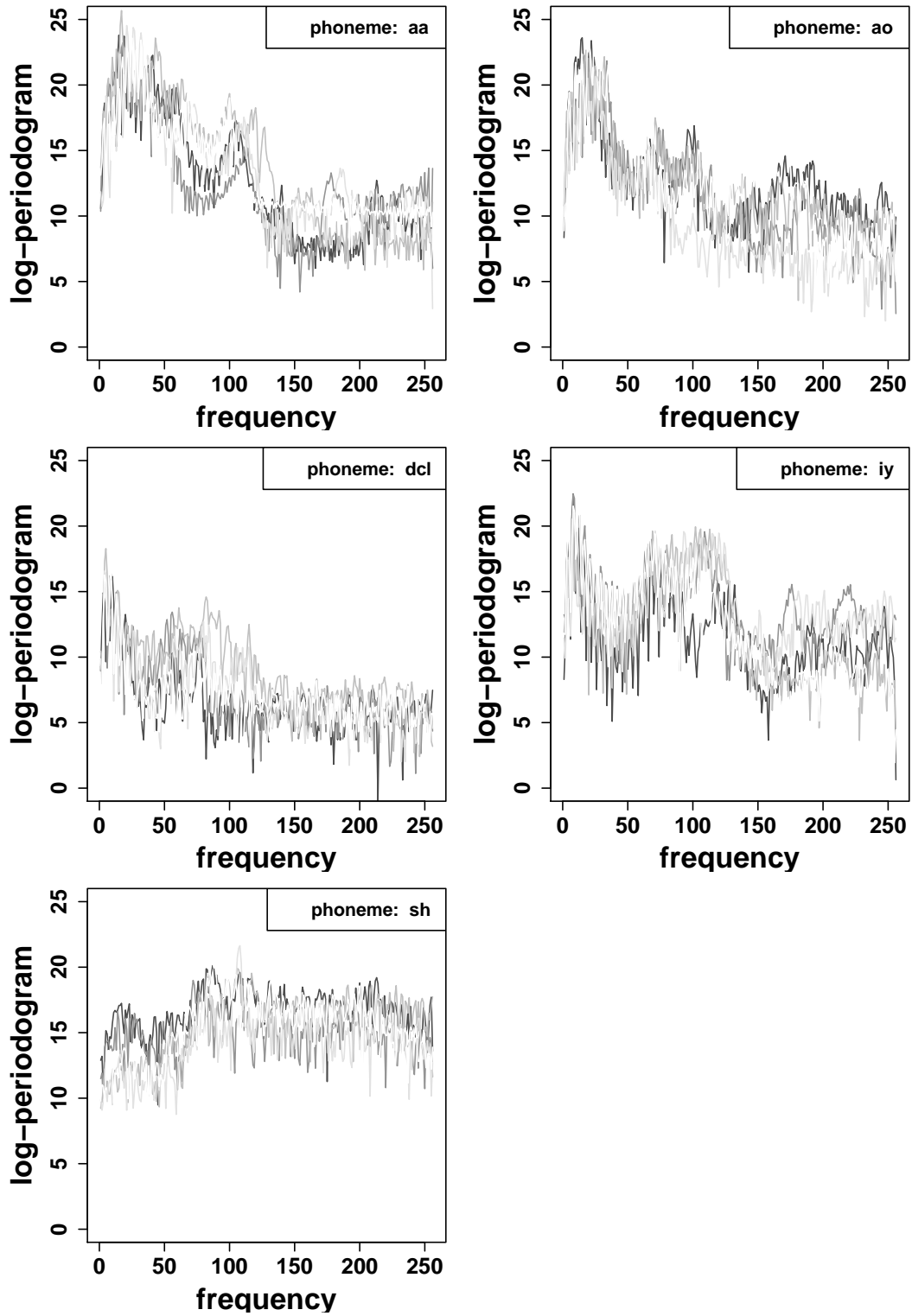


Figure 5: Five exemplarily log-periodograms per phoneme.

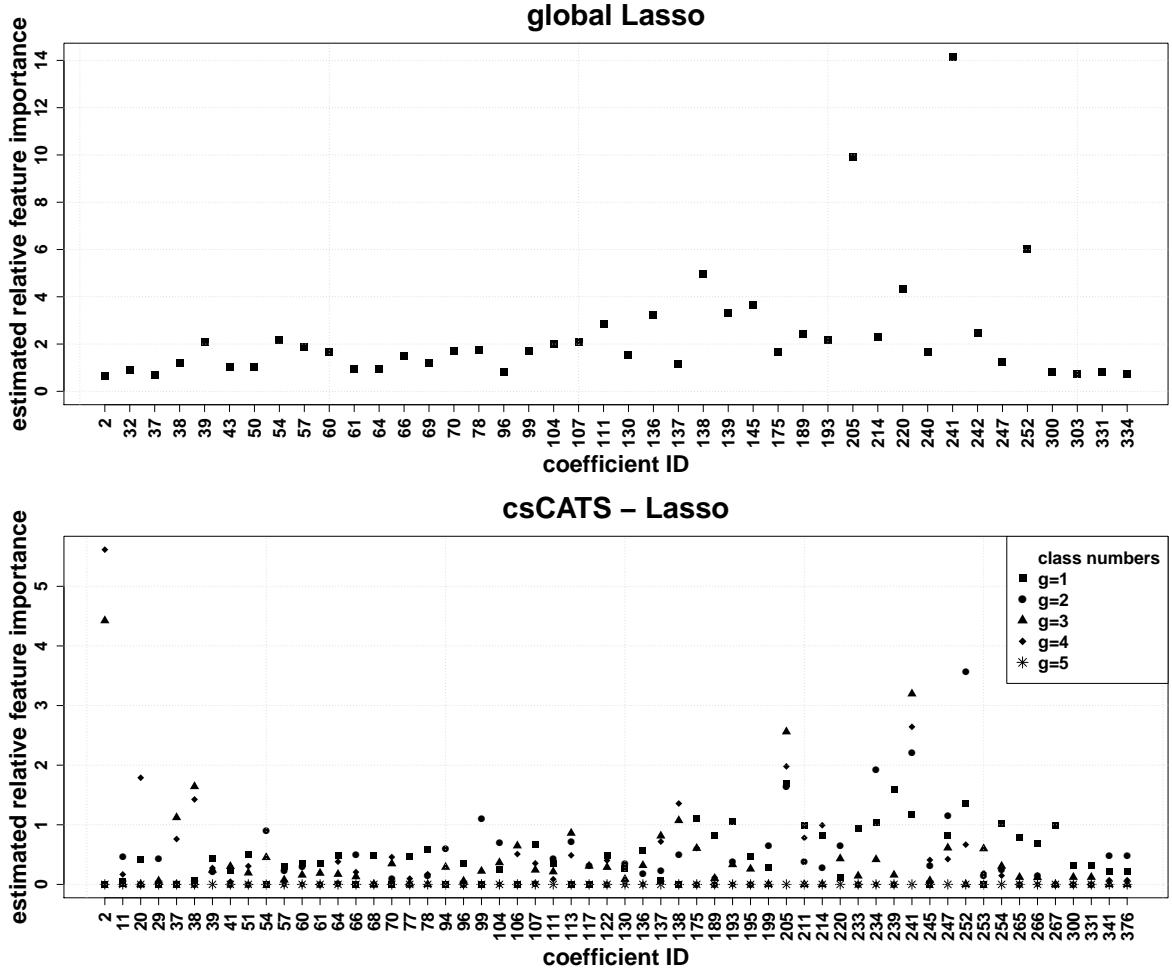


Figure 6: Relative feature importance of the coefficients which have been estimated unequal to zero, as estimated from the whole phoneme data. The upper panel shows the results for the (global) Lasso penalty, the lower panel those for the csCATS penalty.

estimated. Since the data at hand is multi-class data, category-specific penalties might be useful. Thus, we test the performance of the penalized cMLM using the global, the category-specific (cs) and the category-specific CATS (csCATS) penalties introduced in Section 2.2.

In Figures 6 and 7, the symbols present the model coefficients' RFI as estimated from the whole phoneme data, using the respective penalties. For clarity, solely the (Lasso/ cs-Lasso/ csCATS-Lasso) $\hat{=}$ (42/ 101/ 58) coefficients that are estimated to be of values unequal to zero (for at least one category in the case of cs- or csCATS-Lasso) are shown. For the two category-specific penalties, most estimated RFI values vary strongly between the single classes. This indicates that a category-specific penalty is adequate for this data.

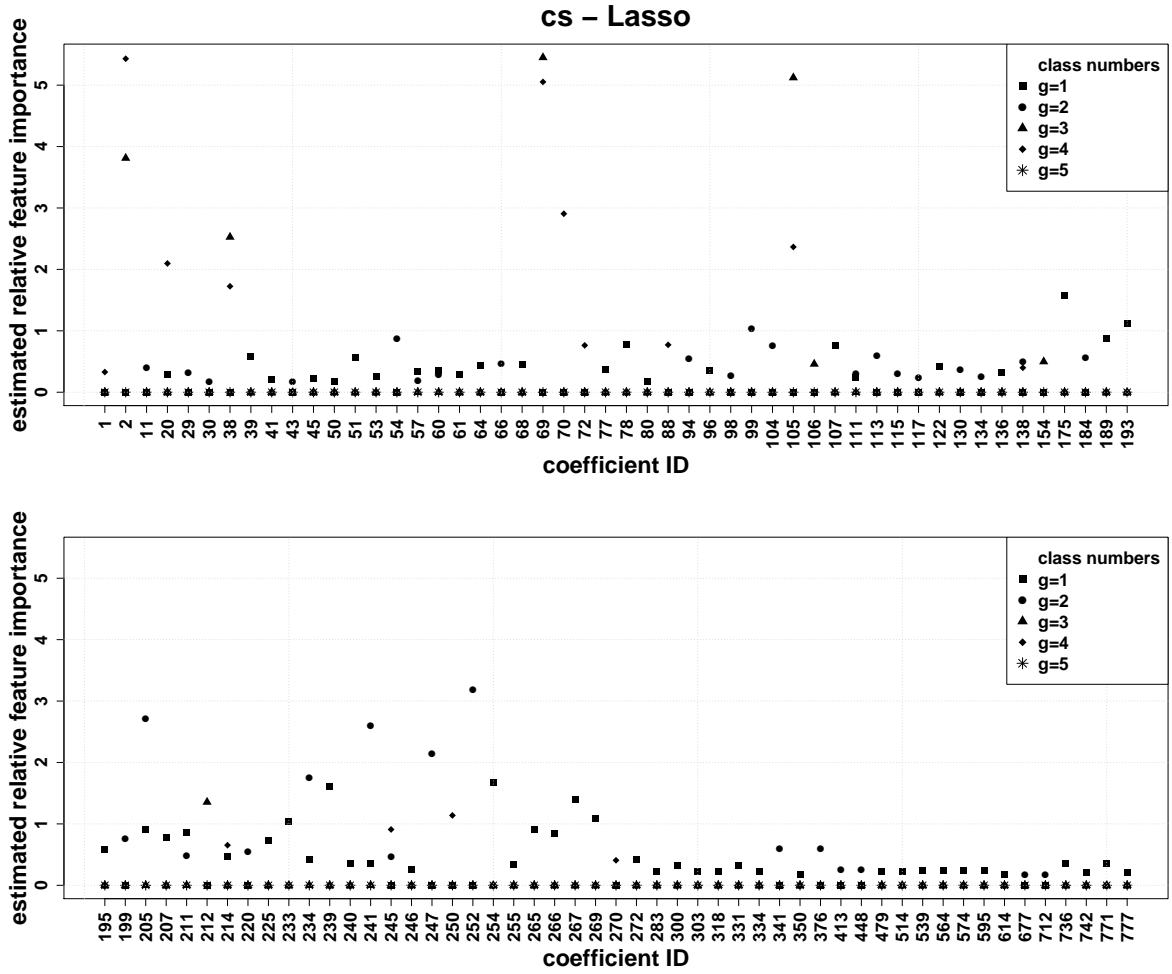


Figure 7: Relative feature importance of the coefficients which have been estimated unequal to zero, as estimated from the whole phoneme data, using the cs-Lasso penalty.

The decoding of the four coefficients with the highest estimated RFI values can be found in Table 4.

To again evaluate the prediction performance of our method and compare it to the other classification methods, 150 curves per class are drawn randomly from the complete data set, being used as a learning data set. The test sample contains another 250 randomly drawn curves per class. The draws, modeling and test steps were repeated 100 times, with all competing methods being applied on identical sample sets.

Again, the estimation results of our approach per draw and penalty type are sparse. Across the 100 replications, the penalized cMLM estimates overall 487 (Lasso)/ 689 (cs-Lasso)/ 315 (csCATS-Lasso) coefficient IDs to have values above zero. However, most coefficients are chosen seldomly, see also the examples in Figures B.10 - D.12 in Appendix B - Appendix D. The coefficient ID's with the four highest estimated mean RFI values are 205, 206, 214, and 241 (Lasso)/ 1, 2, 69, and 70 (cs-Lasso)/ 2, 70, 205, and 241 (csCATS-Lasso), partly overlapping with the coefficients that have been estimated from the whole data shown in Table 4. The tuples corresponding to the unlisted IDs are $1 \hat{=} \{d^{Eucl}, a = 0, k = 1\}$, $70 \hat{=} \{d^{shortEucl}, a = 0, k = 5\}$ with $\mathbb{D}_{small} = [t_1, t_{17}]$, $206 \hat{=} \{d^{shortEucl}, a = 0, k = 21\}$ with $\mathbb{D}_{small} = [t_1, t_{17}]$, and $214 \hat{=} \{d^{relAreas}, a = 0, k = 21\}$ with $\mathbb{D}_1 = [t_{57}, t_{76}]$. It thus seems that the Euclidian distance between the raw or centered curves, eventually using only the very first part of the signal, and with a relatively high number of nearest neighbors $k \geq 11$, is the most important curve characteristic for this discrimination task.

Figure 8 gives the test results for all classification approaches on basis of 100 replications. The upper panel shows the Brier scores (for *RDA*, the estimated probabilities were not accessible), the lower panel the misclassification rates. To achieve a better resolution concerning the best performing methods' boxes, the y-scale has been cut. The mean Brier score of the method SVM - cov. is 0.291. The mean MCRs of SVM - FPCs, *RDA*, and *SDA* are 0.8, 0.81, and 0.95, respectively. As can be seen, the penalized cMLM is competitive compared to the other methods for all penalties, with the lowest Brier scores and MCR values among the best methods. The Brier score also reveals an improvement of prediction accuracy compared to the k -nearest-neighbor ensemble, emphasizing the advantages discussed in Section 2. The performance between the three penalty options Lasso, cs-Lasso and csCATS-Lasso is comparable, except for the cs-Lasso yielding somewhat lower Brier scores than the other penalties.

5. Discussion

We propose a functional classification approach that includes interpretable feature and variable selection by estimating a functional k -nearest-neighbor ensemble within a penal-

coefficient ID (estimated RFI values)	parameter tuple
Lasso	
241 (14.16%)	$\{d^{Eucl} \text{ with } x_i(t) \text{ centered}, a = 0, k = 21\}$
205 (9.92%)	$\{d^{Eucl}, a = 0, k = 21\}$
252 (6.04%)	$\{d^{Max} \text{ with } x_i(t) \text{ centered}, a = 0, k = 21\}$
138 (4.97%)	$\{d^{shortEucl}, a = 0, k = 11\}, \mathbb{D}_{small} = [t_1, t_{17}]$
cs-Lasso	
69 (10.5%)	$\{d^{Eucl}, a = 0, k = 5\}$
2 (9.24%)	$\{d^{shortEucl}, a = 0, k = 1\}, \mathbb{D}_{small} = [t_1, t_{17}]$
105 (7.49%)	$\{d^{Eucl} \text{ with } x_i(t) \text{ centered}, a = 0, k = 5\}$
38 (4.25%)	$\{d^{shortEucl} \text{ with } x_i(t) \text{ centered}, a = 0, k = 1\},$ $\mathbb{D}_{small} = [t_1, t_{17}]$
csCATS-Lasso	
2 (10.04%)	$\{d^{shortEucl}, a = 0, k = 1\}, \mathbb{D}_{small} = [t_1, t_{17}]$
241 (9.22%)	$\{d^{Eucl} \text{ with } x_i(t) \text{ centered}, a = 0, k = 21\}$
205 (7.87%)	$\{d^{Eucl}, a = 0, k = 21\}$
252 (5.58%)	$\{d^{Max} \text{ with } x_i(t) \text{ centered}, a = 0, k = 21\}$

Table 4: Selection results for the single penalties. First column: the IDs of the four estimated coefficients that show the largest RFI values, across all categories where appropriate (in brackets, in decreasing order). Second column: The chosen ensemble coefficients are decoded, with value a indicating the order of derivation and k indicating the number of nearest neighbors used.

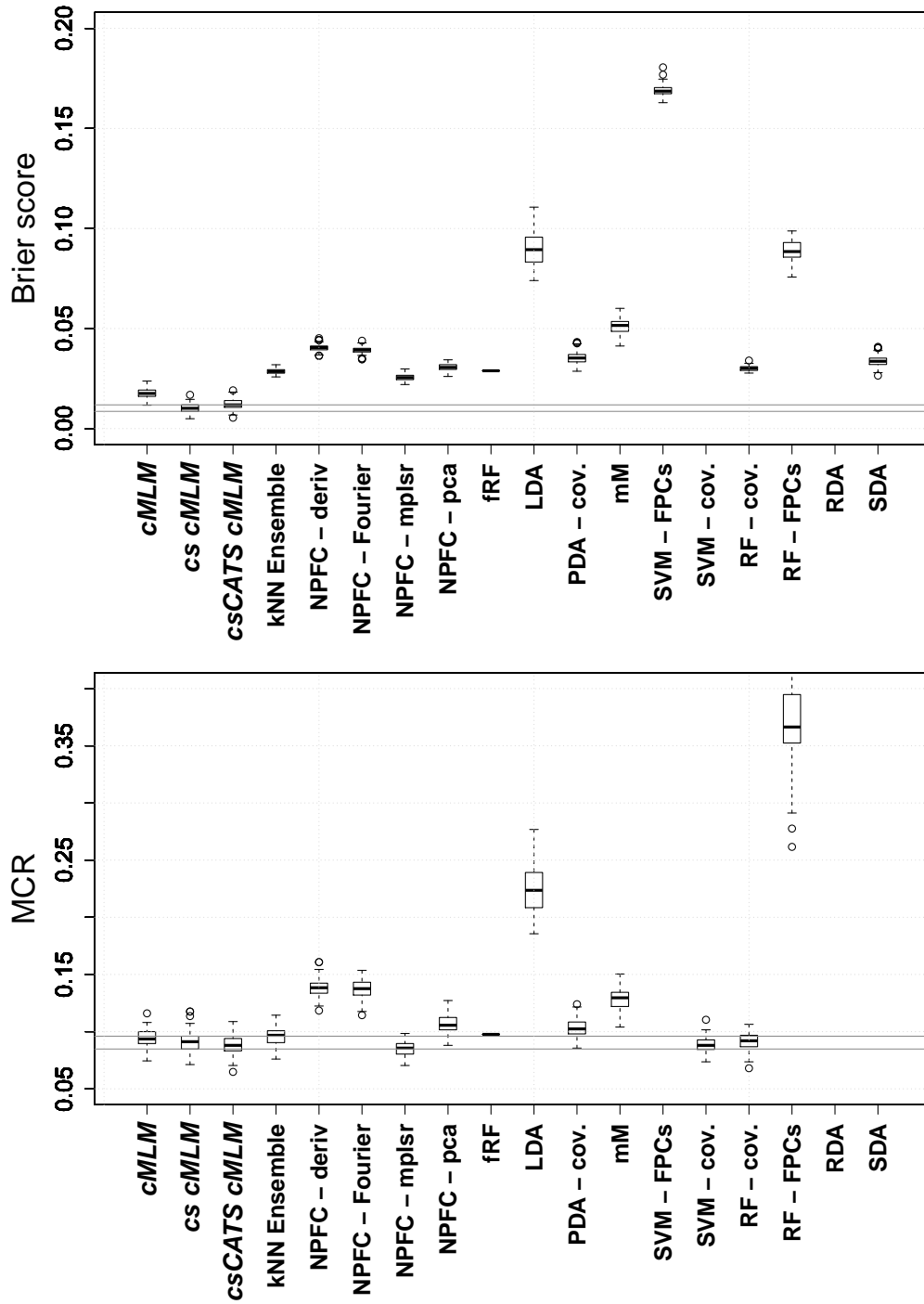


Figure 8: Test results of the phoneme data for all classification approaches on basis of 100 replications. The upper panel shows the Brier scores. The box of method SVM - cov. (mean value 0.291) is not shown due to y-axis pruning. In the lower panel the misclassification rates (MCR) are shown. The boxes of SVM - FPCs, RDA and SDA are not shown due to y-axis pruning. They yield mean values of 0.8, 0.81 and 0.95, respectively. The horizontal lines indicate the values of the first and third quartiles of the category-specific cMLM-box.

ized and constrained multinomial logit model. The approach represents a synthesis of the methods introduced in Tutz et al. (2015) and Fuchs et al. (2015). The strong performance is only obtained by the combination of the methods.

Setting up the functional ensemble can be seen as a dimension reduction approach that allows to detect the relevant features for classification given functional covariates. It makes efficient use of established tools and reliable and fast software. A large variety of penalties enables the user to define additional benefits arising with the discrimination itself. For example, the choice of a Lasso-type penalty results in feature selection. Since the functional k -nearest-neighbor ensemble is estimated within the MLM framework the estimated probabilities π_g of unknown observations $x^*(t)$ are automatically scaled to their natural domain $[0, 1]$, and the constraint $c_l \geq 0$ suffices to make the ensemble coefficients interpretable. This allows the use of category-specific coefficients if necessary. Moreover, the method can readily be adapted to ordinal responses. To summarize, the ensemble step enables a MLM to classify functional covariates, providing inputs that represent a wide variety of curve features. Meanwhile, the MLM step speeds up the estimation of the ensemble weights and allows for more sophisticated coefficient specification. The prediction results of both, the cell chip as well as the phoneme data, support the advantages of the method.

Since the number of data sets including multiple data is growing, the capability of the penalized cMLM to weight differing covariate types differently, up to selecting some types and dispense others, is of increasing interest in the context of functional classification. This was exemplified by the cell chip data. Although both functional covariate types, i.e. the IDEs- and ISFET-signals, were selected, the chosen coefficients represented different curve characteristics. Thus, even if no variable selection takes place, one gets information concerning which features of which signal provide discriminative power, allowing for insight in the processes underlying the data.

We have shown that the estimation of a functional k -nearest-neighbor ensemble via a penalized cMLM is a powerful tool for the discrimination of functional data. Nevertheless, there is ample room for extensions. Concerning the ensemble, an especially worthwhile point to be considered is the choice of the semi-metrics. To be able to achieve data-driven feature selection, a large variety of semi-metrics should be incorporated in the ensemble. The semi-metrics and respective parameters have to be chosen by the user. A randomized choice of both, semi-metrics and semi-metric parameters where appropriate, possibly from predefined sets, seems a sensible enhancement of the k -nearest-neighbor ensemble. Also, multivariate and non-functional covariates could be included by suitable semi-metrics. Concerning the penalized cMLM, the method can handle additional covariates correspond-

ing to (random) batch or time-independent effects by adapting the grouped Lasso penalty. There are also penalties that account for highly correlated inputs, see Tutz and Ulbricht (2009) or Bondell and Reich (2008) for exemplarily penalties applied on non-functional data. However, the combination of all issues mentioned, i.e. ordinal responses, random and time-independent effects as well as highly correlated data, being incorporated into the penalized cMLM approach will probably rise other challenges that could be topics for future research.

Appendix A. Estimated coefficients of the cell chip data across replication splits

In the main article, an important issue was the prediction performance comparison of different methods. To this end, the cell chip data was divided randomly 100 times into learning sets comprising 90 curves and test sets of size 30. While the prediction results of all competing methods were given in Figure 4, the following Figure A.9 gives the cumulated RFI of those penalized cMLM coefficients that are estimated unequal to zero for at least one split, as boxplots across all 100 splits. As can be seen, 84 coefficient IDs across all replications were chosen, and most were selected very seldomly.

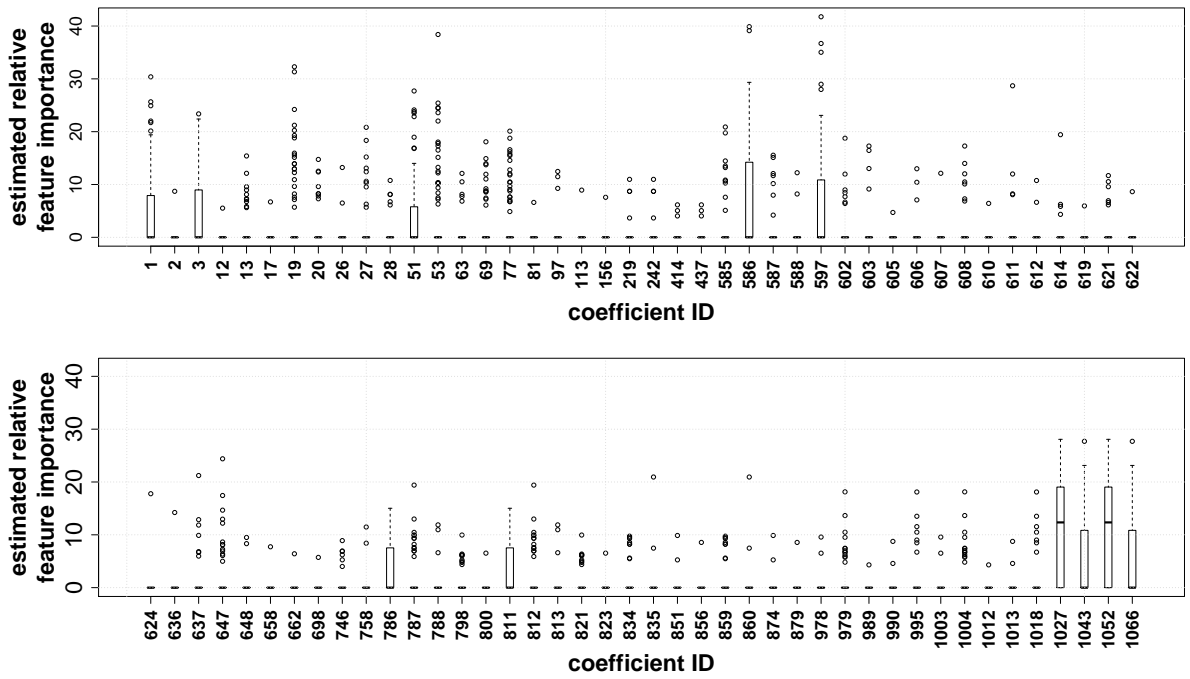
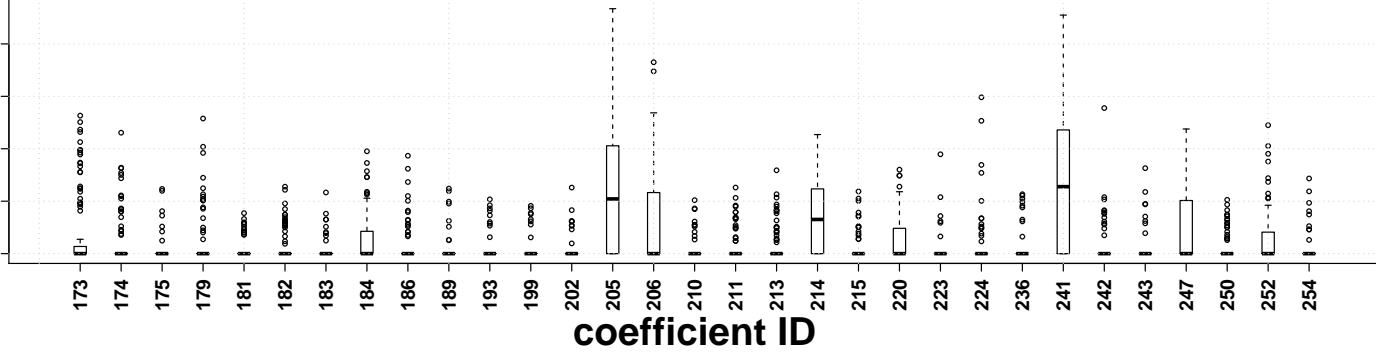


Figure A.9: Coefficients of the cell chip data that are unequal to zero, as boxplots across all 100 splits into learning and test data sets.

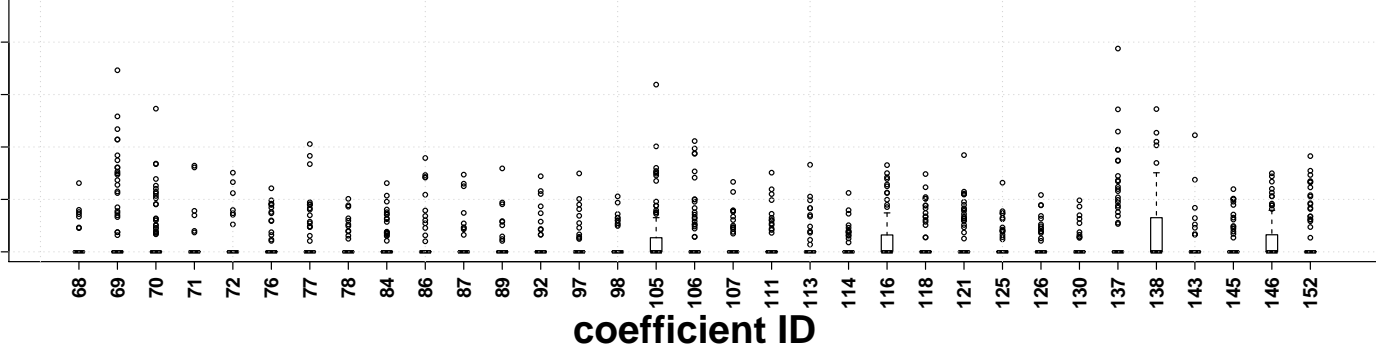
Appendix B. Estimated coefficients of the phoneme data across replication splits

Analogously to the cell chip data, modeling and test steps were repeated 100 times to evaluate the prediction performance of the different classification methods. Here, each random draw used 150 curves per class as a learning data set. The test sample contained another 250 randomly drawn curves per class. The prediction results of all competing methods were given in Figure 8. The following figure gives the cumulated RFI of those globally penalized cMLM coefficients that are estimated to have mean RFI values of above 0.25% (across all splits), as boxplots across all 100 splits. Overall, 487 from 800 coefficient IDs across all replications were chosen, most very seldomly, as, for example, the coefficients 7 - 10 exemplify.

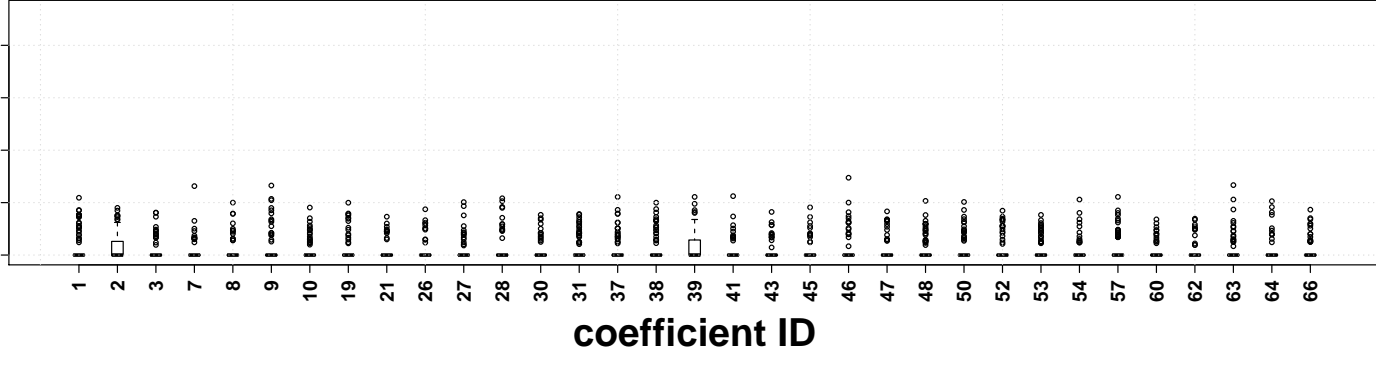
estimated relative
feature importance



estimated relative
feature importance

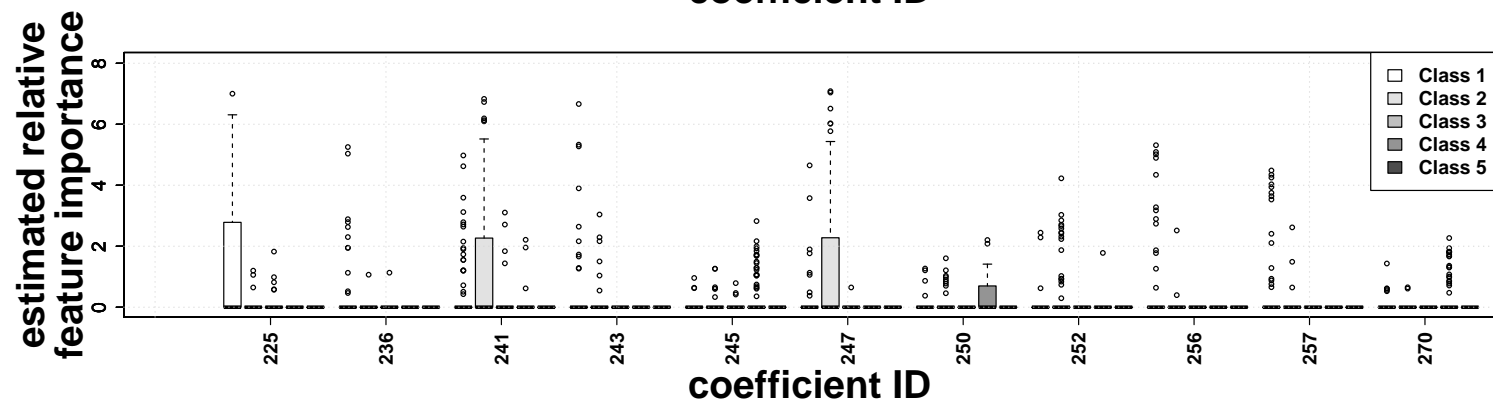
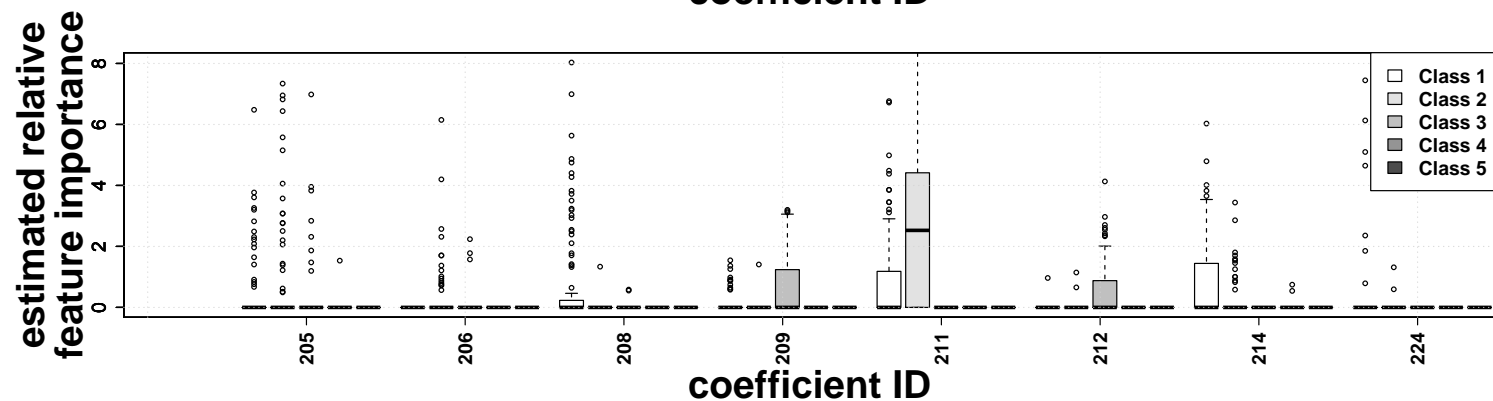
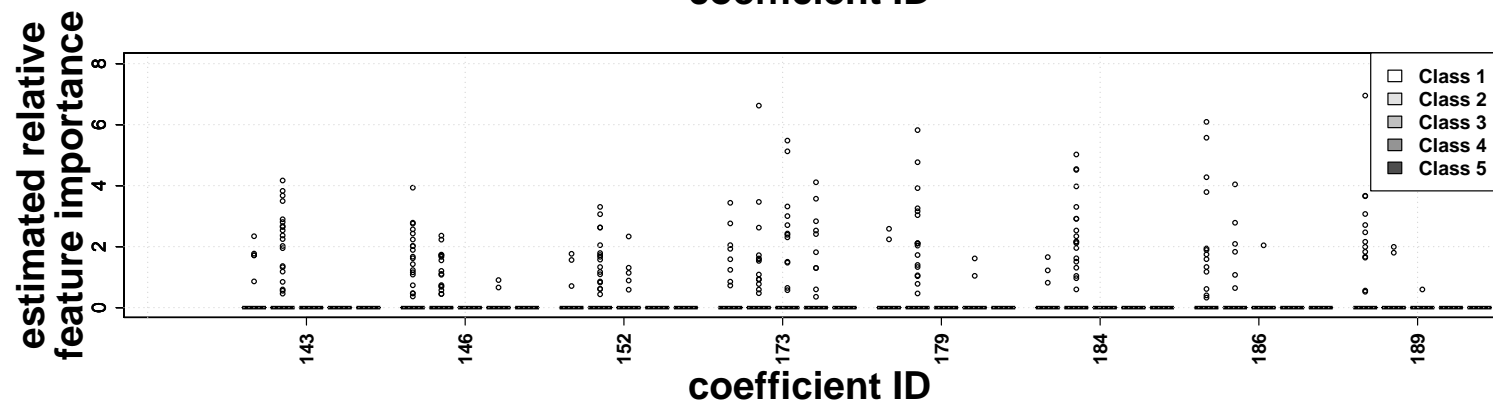
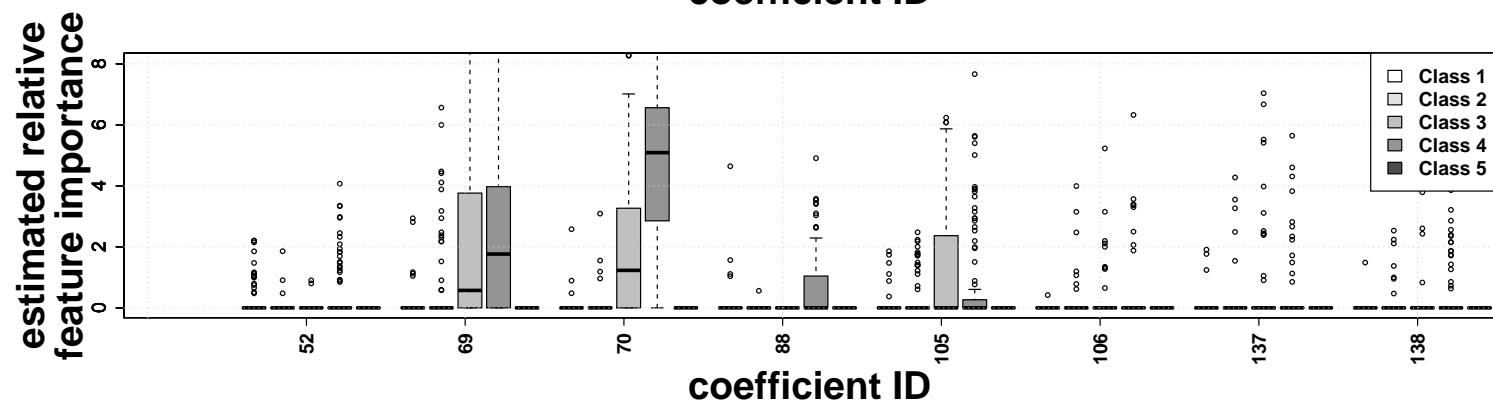
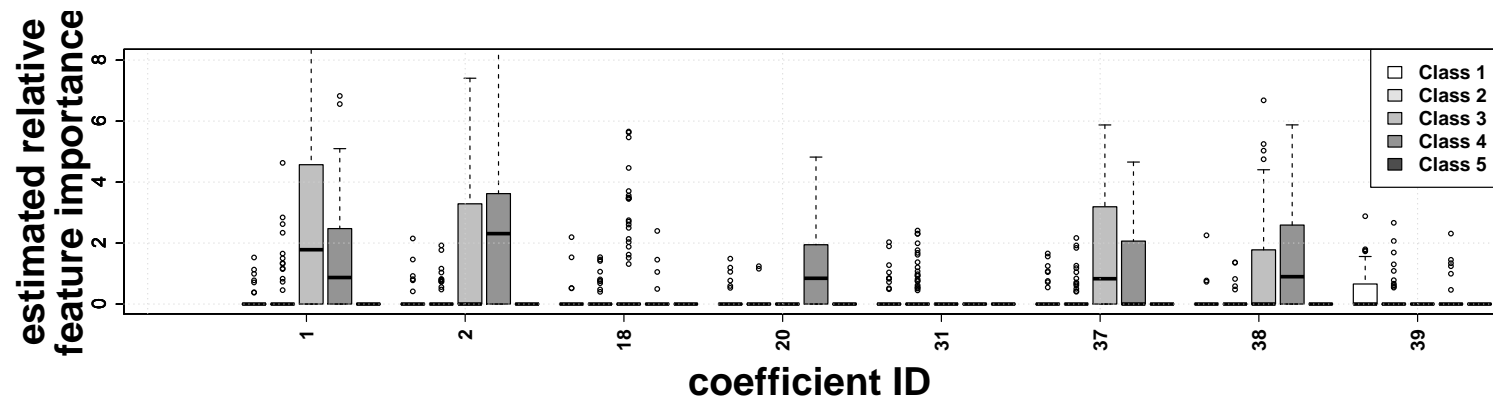


estimated relative
feature importance



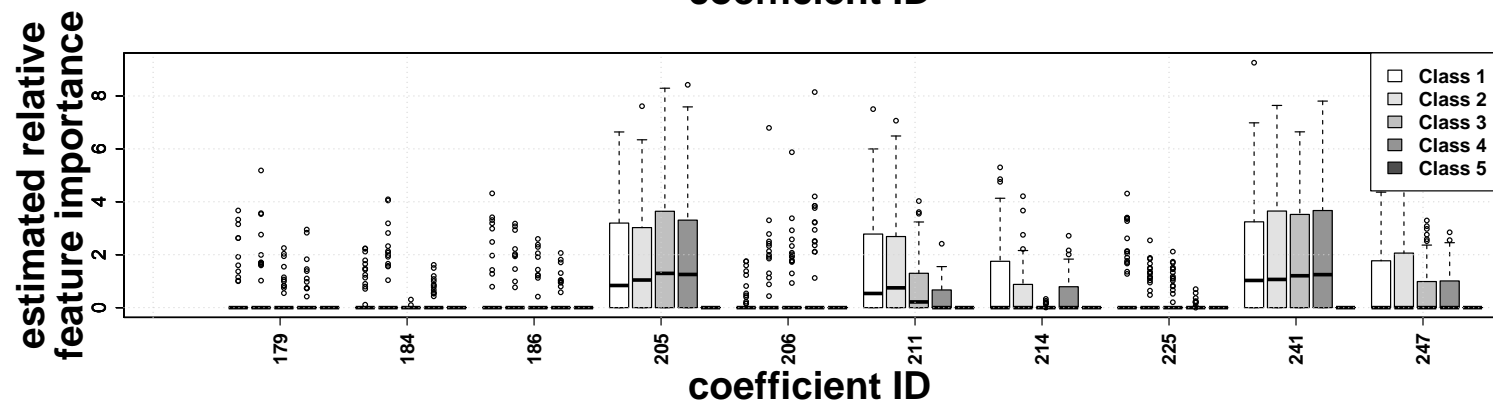
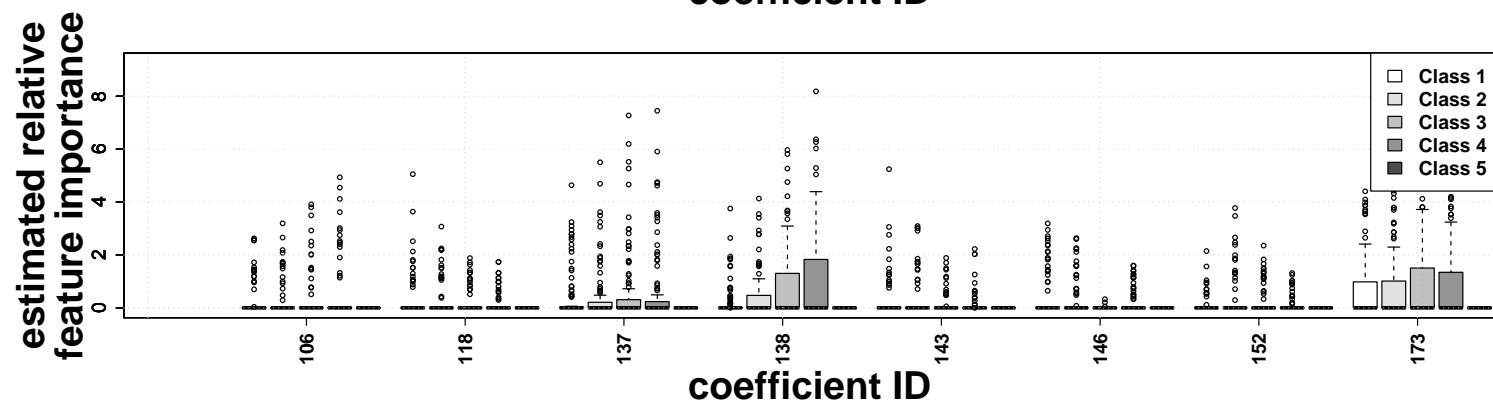
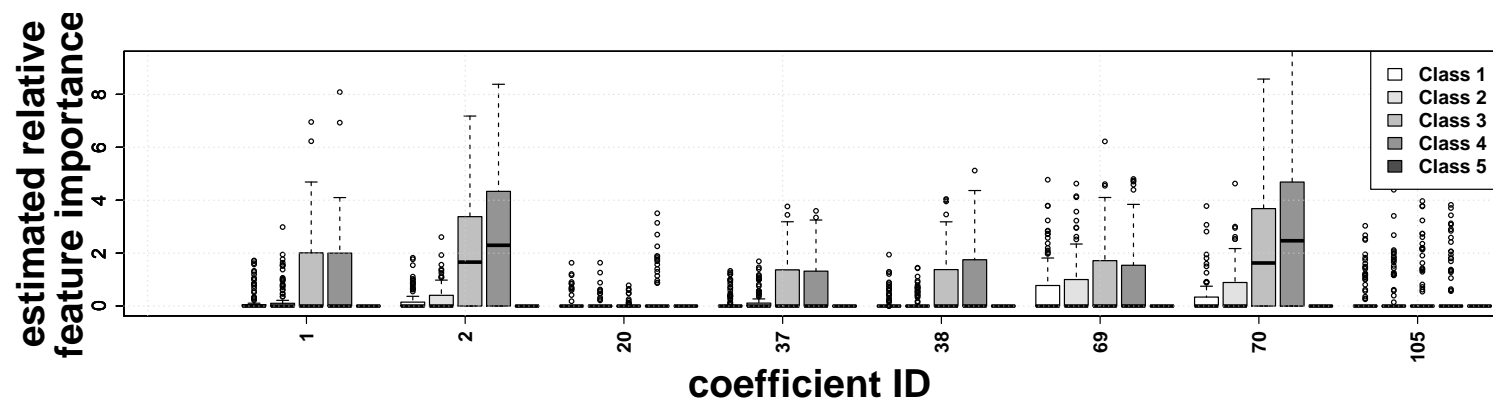
Appendix C. Estimated coefficients of the phoneme data across replication splits, using a category-specific penalty

Analogously to the cell chip data, modeling and test steps were repeated 100 times to evaluate the prediction performance of the different classification methods. Here, each random draw used 150 curves per class as a learning data set. The test sample contained another 250 randomly drawn curves per class. The prediction results of all competing methods were given in Figure 8. The following figure gives the cumulated RFI of those category-specific penalized cMLM coefficients that are estimated to have mean RFI values of above 0.25% (across all splits, for at least one class), as boxplots across all 100 splits. Overall, 689 from 800 coefficient IDs across all replications were chosen, most very seldomly, as, for example, the coefficients 18 or 31 exemplify.



Appendix D. Estimated coefficients of the phoneme data across replication splits, using a category-specific CATS penalty

Analogously to the cell chip data, modeling and test steps were repeated 100 times to evaluate the prediction performance of the different classification methods. Here, each random draw used 150 curves per class as a learning data set. The test sample contained another 250 randomly drawn curves per class. The prediction results of all competing methods were given in Figure 8. The following figure gives the cumulated RFI of those categorically structured (CATS) penalized cMLM coefficients that are estimated to have mean RFI values of above 0.25% (across all splits, for at least one class), as boxplots across all 100 splits. Overall, only 315 from 800 coefficient IDs across all replications were chosen, most very seldomly, as, for example, the coefficients 20 or 105 exemplify.



References

- Ahdesmaki, M., Zuber, V., Gibb, S., and Strimmer, K. (2015). *Shrinkage Discriminant Analysis and CAT Score Variable Selection*, R package version 1.3.7.
- Alonso, A. M., Casado, D., and Romo, J. (2012). Supervised classification for functional data: A weighted distance approach. *Computational Statistics & Data Analysis*, 56:2334 – 2346.
- Araki, Y., Konishi, S., Kawano, S., and Matsui, H. (2009). Functional Logistic Discrimination Via Regularized Basis Expansions. *Communications in Statistics – Theory and Methods*, 38:2944 – 2957.
- Argyriou, A., Evgeniou, T., and Pontil, M. (2007). Multi-task feature learning. In Schölkopf, B., Platt, J., and Hoffman, T., editors, *Advances in Neural Information Processing Systems 19*, pages 41 – 48. MIT Press.
- Athitsos, V. and Sclaroff, S. (2005). Boosting Nearest Neighbor Classifiers for Multiclass Recognition. *IEEE Workshop on Learning in Computer Vision and Pattern Recognition*.
- Bohrn, U., Mucha, A., Werner, C. F., Trattner, B., Bäcker, M., Krumbe, C., Schienle, M., Stütz, E., Schmitt-Landsiedel, D., Fleischer, M., Wagner, P., and Schöning, M. J. (2013). A critical comparison of cell-based sensor systems for the detection of Cr(VI) in aquatic environment. *Sensors and Actuators, B: Chemical*, 182:58 – 65.
- Bohrn, U., Stütz, E., Fuchs, K., Fleischer, M., Schöning, M. J., and Wagner, P. (2012). Monitoring of irritant gas using a whole-cell-based sensor system. *Sensors and Actuators, B: Chemical*, 175:208 – 217.
- Bondell, H. D. and Reich, B. J. (2008). Simultaneous Regression Shrinkage, Variable Selection, and Supervised Clustering of Predictors with OSCAR. *Biometrics*, 64:115 – 123.
- Breiman, L. (2001). Random Forests. *Machine Learning*, 45:5 – 32.
- Breiman, L., Cutler, A., Liaw, A., and Wiener, M. (2012). *randomForest : Breiman and Cutler's random forests for classification and regression*, R package version 4.6-7.
- Brier, G. W. (1950). Verification of forecasts expressed in terms of probability. *Monthly Weather Review*, 78 (1):1 – 3.

- Cerioti, L., Kob, A., Drechsler, S., Ponti, J., Thedinga, E., Colpo, P., Ehret, R., and Rossi, F. (2007). Online monitoring of BALB/3T3 metabolism and adhesion with multiparametric chip-based system. *Analytical Biochemistry*, 371:92 – 104.
- Chen, J. and Li, H. (2013). Variable selection for sparse Dirichlet-Multinomial regression with an application to microbiome data analysis. *The Annals of Applied Statistics*, 7(1):418 – 442.
- Clemmensen, L., Hastie, T., Witten, D., and Ersboll, B. (2011). Sparse Discriminant Analysis. *Technometrics*, 53 (4):406–413.
- Crainiceanu, C., Reiss, P., Goldsmith, J., Huang, L., Huo, L., Scheipl, F., Swihart, B., Greven, S., Harezlak, J., Kundu, M. G., Zhao, Y., McLean, M., and Xiao, L. (2013). *refund : Regression with Functional Data*, R package version 0.1-9.
- Delaigle, A. and Hall, P. (2012). Achieving near perfect classification for functional data. *Journal of the Royal Statistical Society: Series B*, 74 (2):267 – 286.
- Di, C., Crainiceanu, C. M., Caffo, B. S., and Punjabi, N. M. (2009). Multilevel functional principal component analysis. *The Annals of Applied Statistics*, 3(1):458 – 488.
- Epifanio, I. (2008). Shape Descriptors for Classification of Functional Data. *Technometrics*, 50(3):284 – 294.
- Ferraty, F. and Vieu, P. (2003). Curves discrimination: a nonparametric functional approach. *Computational Statistics & Data Analysis*, 44:161 – 173.
- Ferraty, F. and Vieu, P. (2006). *Nonparametric Functional Data Analysis*. Springer Science and Business Media, New York.
- Fisher, R. A. (1936). The use of multiple measurements in taxonomic problems. *Annals of Eugenics*, pages 179 – 188.
- Fuchs, K., Gertheiss, J., and Tutz, G. (2015). Nearest neighbor ensembles for functional data with interpretable feature selection. *Chemometrics and Intelligent Laboratory Systems*, 146:186 – 197.
- Gertheiss, J., Maity, A., and Staicu, A.-M. (2013). Variable selection in generalized functional linear models. *Stat*, 2:86 – 101.
- Gertheiss, J. and Tutz, G. (2009). Feature selection and weighting by nearest neighbor ensembles. *Chemometrics and Intelligent Laboratory Systems*, 99:30 – 38.

- Gneiting, T. and Raftery, A. E. (2007). Strictly Proper Scoring Rules, Prediction, and Estimation. *Journal of the American Statistical Association*, 102 (477):359 – 378.
- Goldsmith, J., Greven, S., and Crainiceanu, C. (2013). Corrected Confidence Bands for Functional Data Using Principal Components. *Biometrics*, 69 (1):41 – 51.
- Guo, Y., Hastie, T., and Tibshirani, R. (2007). Regularized linear discriminant analysis and its application in microarrays. *Biostatistics*, 8 (1):86–100.
- Guo, Y., Hastie, T., and Tibshirani, R. (2012). *Shrunken Centroids Regularized Discriminant Analysis*, R package version 1.0.2-2.
- Hall, P., Poskitt, D. S., and Presnell, B. (2001). A Functional Data-Analytic Approach to Signal Discrimination. *Technometrics*, 43 (1):1 –9.
- Hastie, T., Buja, A., and Tibshirani, R. (1995). Penalized Discriminant Analysis. *The Annals of Statistics*, 23 (1):73 – 102.
- Hastie, T., Tibshirani, R., Friedman, J., and Halvorsen, K. (2015a). *ElemStatLearn : Data sets, functions and examples from the book : “The Elements of Statistical Learning, Data Mining, Inference, and Prediction” by Trevor Hastie, Robert Tibshirani and Jerome Friedman*, R package version 2012-04-05.
- Hastie, T., Tibshirani, R., Leisch, F., Hornik, K., and Ripley, B. D. (2015b). *mda : Mixture and flexible discriminant analysis*, R package version 0.4-4.
- James, G. M. (2001). Functional linear discriminant analysis for irregularly sampled curves. *J. R. Statist. Soc. B*, 63 (3):533 – 550.
- Jenatton, R., Mairal, J., Obozinski, G., and Bach, F. (2011). Proximal methods for hierarchical sparse coding. *Journal of Machine Learning Research*, 12:2297–2334.
- Jurka, T. P. and Tsuruoka, Y. (2013). *maxent : Low – memory Multinomial Logistic Regression with Support for Text Classification*, R package version 1.3.3.1.
- Kubisch, R., Bohrn, U., Fleischer, M., and Stütz, E. (2012). Cell-based sensor system using L6 cells for broad band continuous pollutant monitoring in aquatic environments. *Sensors*, 12:3370 – 3393.
- Li, B. and Yu, Q. (2008). Classification of functional data: A segmentation approach. *Computational Statistics & Data Analysis*, 52:4790 – 4800.
- Matsui, H. (2014). Variable and boundary selection for functional data via multiclass logistic regression modeling. *Computational Statistics & Data Analysis*, 78:176 – 185.

- McFadden, D. (1973). Conditional logit analysis of qualitative choice behaviour. In *Frontiers in Econometrics*. ed. by P. Zarembka, Academic Press, New York.
- Meyer, D., Dimitriadou, E., Hornik, K., Weingessel, A., Leisch, F., Chang, C.-C., and Lin, C.-C. (2014). *e1071 : Misc Functions of the Department of Statistics (e1071)*, TU Wien, R package version 0.1-9.
- Möller, A., Tutz, G., and Gertheiss, J. (2016). Random forests for functional covariates. *Journal of Chemometrics*, *accepted for publication*.
- Müller, H. G. and Stadtmüller, U. (2005). Generalized Functional Linear Models. *The Annals of Statistics*, 33 (2):774 – 805.
- Nguyen, H. D., McLachlan, G. J., and Wood, I. A. (2016). Mixtures of spatial spline regressions for clustering and classification. *Computational Statistics and Data Analysis*, 93:76 – 85.
- Pößnecker, W. (2015). *MRSP : Multinomial Response Models with Structured Penalties*, R package version 0.4.3.
- Preda, C., Saporta, G., and Leveder, C. (2007). PLS classification of functional data. *Computational Statistics*, 22:223 – 235.
- R Core Team (2016). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria.
- Ramsay, J. and Silverman, B. (2005). *Functional Data Analysis*. Springer: New York.
- Ramsay, J. O. and Silverman, B. W. (2002). *Applied functional data analysis*. Springer-Verlag Inc., New York.
- Rao, C. R. (1973). *Linear Statistical Inference and Its Applications*. John Wiley & Sons Inc.
- Ripley, B., Venables, B., Bates, D. M., Hornik, K., Gebhardt, A., and Firth, D. (2014). *MASS : Support Functions and Datasets for Venables and Ripley's MASS*, R package version 7.3-30.
- Rossi, F. and Villa, N. (2006). Support vector machine for functional data classification. *Neurocomputing*, 69:730 – 742.
- Selten, R. (1998). Axiomatic Characterization of the Quadratic Scoring Rule. *Experimental Economics*, 1:43 – 62.

- Simon, N., Friedman, J., and Hastie, T. (2013). A blockwise descent algorithm for group-penalized multiresponse and multinomial regression. *arXiv preprint*.
- Thedinga, E., Kob, A., Holst, H., Keuer, A., Drechsler, S., Niendorf, R., Baumann, W., Freund, I., Lehmann, M., and Ehret, R. (2007). Online monitoring of cell metabolism for studying pharmacodynamic effects. *Toxicology and Applied Pharmacology*, 220:33 – 44.
- Tibshirani, R. (1996). Regression Shrinkage and Selection via the Lasso. *J. R. Statist. Soc. B*, 58 (1):267 – 288.
- Tutz, G. (2012). *Regression for categorical data*. Cambridge University Press.
- Tutz, G., Pößnecker, W., and Uhlmann, L. (2015). Variable selection in general multinomial logit models. *Computational Statistics & Data Analysis*, 82:207 – 222.
- Tutz, G. and Ulbricht, J. (2009). Penalized regression with correlation-based penalty. *Statistical Computing*, 19:239 – 253.
- Vapnik, V. (1996). *The Nature of Statistical Learning Theory*. Springer, New York.
- Vincent, M. and Hansen, N. (2014). Sparse group lasso and high dimensional multinomial classification. *Computational Statistics & Data Analysis*, 71:771 – 786.
- Wood, S. (2014). *mgcv : Mixed GAM Computation Vehicle with GCV/ AIC/ REML Smoothness Estimation*, R package version 1.8-4.
- Zhu, H., Vannucci, M., and Cox, D. D. (2010). A Bayesian Hierarchical Model for Classification with Selection of Functional Predictors. *Biometrics*, 66:463 – 473.